

DESIGN OF TRAJECTORIES USING INTRINSIC GEOMETRY

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

by

N. SREENIVAS RAO

to the

DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY KANPUR

APRIL, 1992

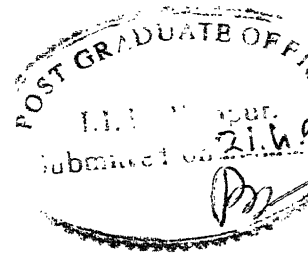
1 8 MAY 1992

CENTRAL LIBRARY
111 KANPUR

Acc. No. A.113463

ME-1882-M-RAC-DES

CERTIFICATE



This is to certify that the present work entitled, "Design Trajectories Using Intrinsic Geometry" has been carried out under my supervision and that it has not been submitted elsewhere for the award of a degree.

Dr. S.G.Dhande

Professor,
Mechanical Engineering and
Computer Science & Engineering
I.I.T.Kanpur.

April 20, 1992.

TABLE OF CONTENTS

CERTIFICATE

TABLE OF CONTENTS

ACKNOWLEDGMENT

LIST OF FIGURES

LIST OF SYMBOLS

ABSTRACT

Chapter

1 INTRODUCTION

- 1.1 Trajectory Planning in Robotics
- 1.2 Design of Planar and Spatial Curves
- 1.3 Literature Review
- 1.4 Objectives and Scope of Present Work

2 INTRINSIC GEOMETRY AND CURVE DESIGN

- 2.1 Basic Concepts of Intrinsic Geometry
- 2.2 Planar Curve Design
- 2.3 Planar Trajectory Design

3 SPATIAL TRAJECTORY PLANNING

- 3.1 Methodology of Three-dimensional Curve Design

3.2 Spatial Trajectory Design

3.3 Application of Spatial Trajectories

4 CASE STUDIES AND EXAMPLES

4.1 Implementation Details

4.2 Examples of 2-D

4.3 Examples of 3-D

4.4 Observation on Numerical Work

5 CONCLUSIONS

5.1 Technical Summary

5.2 Recommendations for Further Work

REFERENCES

ACKNOWLEDGEMENTS

I express my sincerest gratitude to my guide Dr. S.G. Dhande for the untiring guidance and unceasing encouragement without which this work wouldn't have been possible. Here I must also say that working with him has been an unforgettable experience.

I would like to thank all my friends who made my stay at IIT Kanpur pleasant and enjoyable. In this connection I would make special mention of Subba Rao, Piyush, Anand Singh, Reddy (N V, Amarnath, N S), Jones, Sridhar and Venkatesh. My special Thanks to Anand Singh for helping me in preparation of thesis.

Facilities Available in CAD-Project laboratory made my work easy and enjoyable. I would like to thank the staff of CAD-Project for their co-operation.

Finally, I sincerely thank my parents, without whose full support, this thesis would not have been possible.

N. Sreenivas Rao

LIST OF FIGURES

Number	Title	Page No
1.1	Trajectory Planner Block Diagram	2
1.2	Position Conditions for a Joint Trajectory	5
1.3	Tri-hedron Showing Tangent, Normal and Binormal of a Three-dimensional Curve	13
2.1	Intrinsic Geometry of a Three-dimensional Curve	20
2.2	Two-dimensional Curve Design Using λ -s Plane	22
2.3	S-C-S Shape Model with Positive Change in Tangent Angle	27
2.4	S-C-S Shape Model with Negative Change in Tangent Angle	27
2.5	C-S-C Shape Model with Positive Tangent Angle Change for both the Clothoid Pairs	28
2.6	C-S-C Shape Model with Positive Tangent Angle Change for the first Clothoid Pair and Negative for the second	28
2.7	C-S-C Shape Model with Negative Tangent Angle Change for the first Clothoid Pair and Positive for the second	29
2.8	C-S-C Shape Model with Negative Tangent Angle Change for both the Clothoid Pairs	29
2.9	Cartesian Space of a S-C-S Shape Model Planar Curve	31
2.10	C-S-C Shape Model with Positive Change in Tangent Angle	31

2.11	Variation of Sharpness Vs. arc length for S-C-S Shape Model	32
2.12	Cartesian Space of a C-S-C Shape Model Planar Curve	35
3.1	A Schematic Diagram of Input Conditions, Skew and Base Curve	43
3.2	Clothoid Blended Curve with a Straight Line Portion in between to Model the Rise of a 3-D Curve	46
3.3	Clothoid Blended Curve without a Straight Line Portion in between to Model the Rise of a 3-D Curve	46
3.4	Cubically Blended Curve for Modeling the Rise of 3-D Curve	48
4.1	Cartesian Plot of Cornu Spirals with different sharpness	59
4.2	Example of S-C-S Shape Model Curve with Positive Change in Tangent Angle	61
4.3	Example of S-C-S Shape Model Curve with Negative Change in Tangent Angle	62
4.4	Example of C-S-C Shape Model Curve with Positive Change in Tangent Angle for the First Clothoid Pair and Negative for the Second	64
4.5	Example of C-S-C Shape Model Curve with Negative Change in Tangent Angle for the First Clothoid Pair and Positive for the Second	65
4.6	Example of C-S-C Shape Model Curve with Positive Change in Tangent Angle for the Both the Clothoid Pairs	67
4.7	Example of C-S-C Shape Model Curve with Negative Change in Tangent Angle for the Both the Clothoid Pairs	68
4.8	Velocity Plots for C-S-C Shape Model Curve with	70

$ds/dt = \text{constant}$

4.9	Acceleration plots for C-S-C Shape Model Curve with $ds/dt = \text{constant}$	71
4.10	Isometric view of C-S-C Shape Model Curve with Cubically Blended Rise Curve	73
4.11	U-V and W-S' Plots of the 3-D Curve with C-S-C Shape Model Base Curve and Cubically Blended Rise Curve	74
4.12	Velocity and Acceleration Plots of the Z-Coordinate when the Rise Curve is Cubically Blended	75
4.13	Velocity Plots for 3-D Curve Example with C-S-C Shape Model for Base Curve in x and y directions	76
4.14	Acceleration Plots for 3-D Curve Example with C-S-C Shape Model for Base Curve in x and y directions	77

LIST OF SYMBOLS

\mathbf{b}	Binormal vector at a generic point P .
C-S-C	Clothoid - Straight line - Clothoid.
\mathbf{m}_1	Projection of the tangent vector \mathbf{t}_A in the O_1 - UV Plane.
\mathbf{m}_2	Projection of the tangent vector \mathbf{t}_B in the O_1 - UV Plane.
\mathbf{n}	Normal vector at a generic point P .
\mathbf{r}	Position vector of a point.
s	Arc length from a reference point to a generic point measured along the curve.
S-C-S	Straight line - Clothoid - Straight line.
t	Time
\mathbf{t}	Tangent vector at a generic point P .
x,y,z	Cartesian coordinates of a generic point P .
w	Rise of a helix.
κ	Curvature of a generic point P .
τ	Torsion of a generic point P .
θ	Tangent angle.
Θ	Change in tangent angle of the clothoid pair.
γ	Tangent angle of a cubically blended curve.
λ	Sharpness of the clothoid curve.
σ	Dummy variable.

ABSTRACT

The present work outlines a methodology of shape synthesis using the concept of intrinsic geometry, for the design of two-dimensional and three-dimensional curves. Using the concepts of intrinsic geometry the shape of a curve can be defined in terms of intrinsic parameters such as curvature and torsion as a function of arc length. The method of shape synthesis proposed for two-dimensional curves consists of selecting the shape model, defining the shape design variable and then evaluating the cartesian coordinates of the curve. It is assumed that the end point coordinates as well as the tangents are specified. The shape model is conceived as a set of symmetric pair of continuous piecewise linear curvature segments, each segment being defined as a function of arc length.

The shape design of a three dimensional curve is accomplished by modeling it with the help of two planar curves. The first planar curve is the base of the helix defined in its intrinsic form as mentioned above. The second curve defined in the skew direction of the two tangents is the rise curve of the helix defined either as a cubically blended curve with a straight line portion in between, or in its intrinsic form. The three-dimensional curve is finally defined in parametric form using arc length of the base curve as the parameter.

The proposed method has been shown to be useful for designing two-dimensional or three-dimensional path of a mobile robot or the trajectory of the manipulator arm. Finally an illustrative example of the trajectory planning has been described in Chapter 4.

CHAPTER 1

INTRODUCTION

The need for flexibility, improved productivity, better quality and lower production cost in industries has motivated the speedy growth of automation in manufacturing processes. Special purpose machines were the first step towards this automation. Development of NC and CNC machine tools was the next major event in automation in manufacturing processes. The developments which are now taking place in Robotics will ultimately usher in an era of automatic factories.

An industrial robot is a multi-functional automatic computer-controlled mechanical manipulator capable of performing a variety of tasks through motions along several directions. Robot manipulators need to be controlled in efficient and effective ways to enable them to perform the multitude of complex tasks for which they are designed. So, trajectory planning and motion control has evolved as an important research area.

1.1 Trajectory Planning in Robotics

A systematic approach to the trajectory planning problem is to view the trajectory planner as a black box as shown in Figure 1.1. The trajectory planner accepts the input variables which

satisfies the constraints at interpolation points. In this approach the constraint specification and the planning of trajectory are performed in the joint coordinates. Since no constraints are imposed on the manipulator hand, it is difficult to track the path the manipulator hand traverses, hence in this method the manipulator hand may hit obstacles with no prior warning. In the second approach the user explicitly specifies the path which the manipulator hand has to traverse by an analytical function, such as a straight line path in cartesian coordinates, and the trajectory planner determines the desired trajectory either in joint coordinates or in cartesian coordinates that approximate the path.

Trajectory planning can be done either in joint-variable space or in the cartesian space. For joint-variable space planning, the time history of all the joint variables and their first two derivatives are planned to describe the desired motion of the manipulator. For cartesian space planning, the time history of manipulator hand's position, velocity, and acceleration are planned and the corresponding joint positions, velocities and accelerations are derived from the hands information. The joint-variable space planning has following three advantages

- The trajectory planned is directly in terms of controlled variables.
- Trajectory planning can be done in near real time
- The joint trajectories are easier to plan.

The associated disadvantage is the difficulty in determining the locations of the various links and the hand during motion, a task that is usually required to guarantee obstacle avoidance along the trajectory.

Joint Interpolated Trajectories:

While planning joint interpolated trajectories the following considerations are of interest (Fu, Gonzalez and Lee, 1987).

- (i). When picking up an object, the motion of the hand must be directed away from an object, otherwise the hand may crash into the supporting surface of the object.
- (ii) If we specify a departure position (lift off point) along the normal vector to the surface out from the initial point and if we require the hand to pass through this position, we then have an admissible departure motion. If we further specify the time required to reach the position, we can also control the speed at which the object is to be lifted.
- (iii) The same set of lift-off requirements for the arm motion is also true for the set down point of the final position of motion so that the correct approach direction can be obtained and controlled.
- (iv) From the above mentioned points we have four positions for each arm motion as shown in Figure 1.2.: initial, lift-off, set-down and final.
- (v) Constraints:

- (a) Initial position: velocity and acceleration are given and

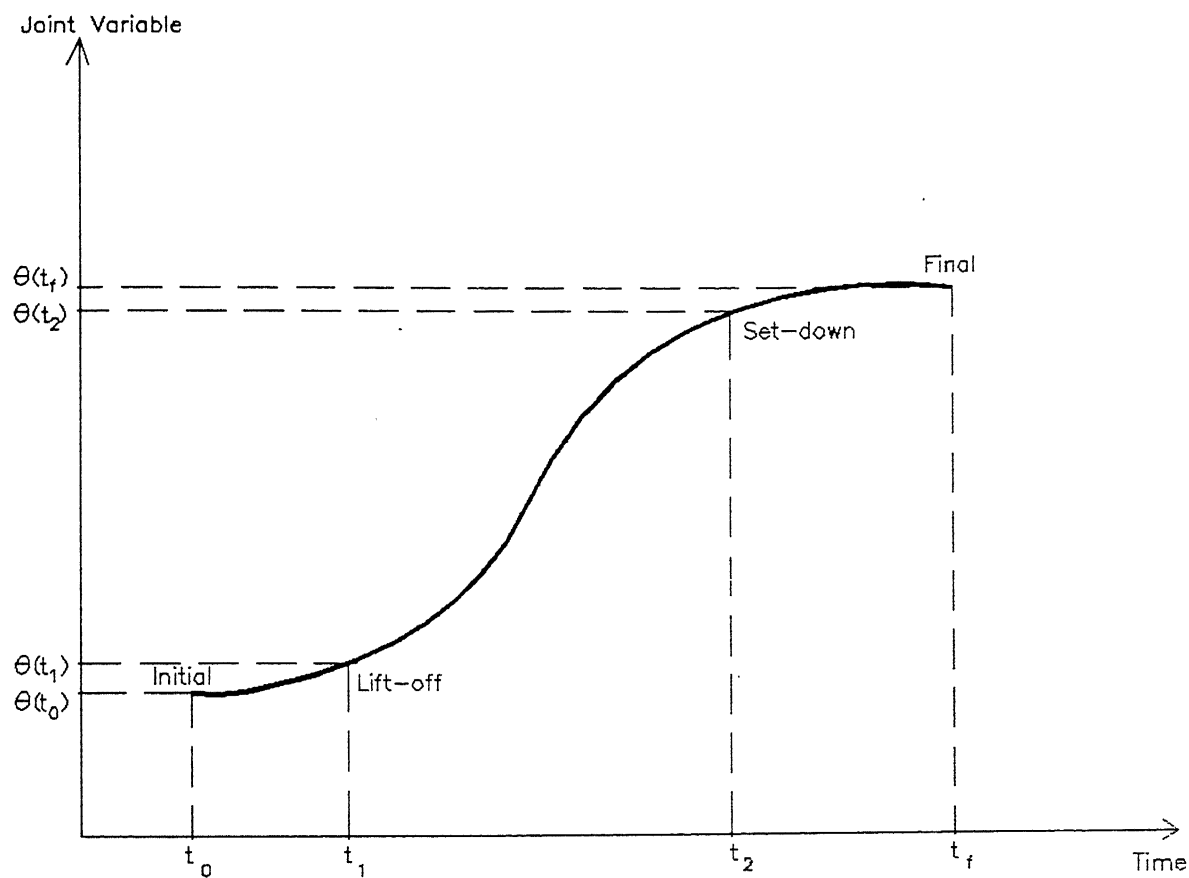


Figure 1.2 Position Condition for a Joint Trajectory

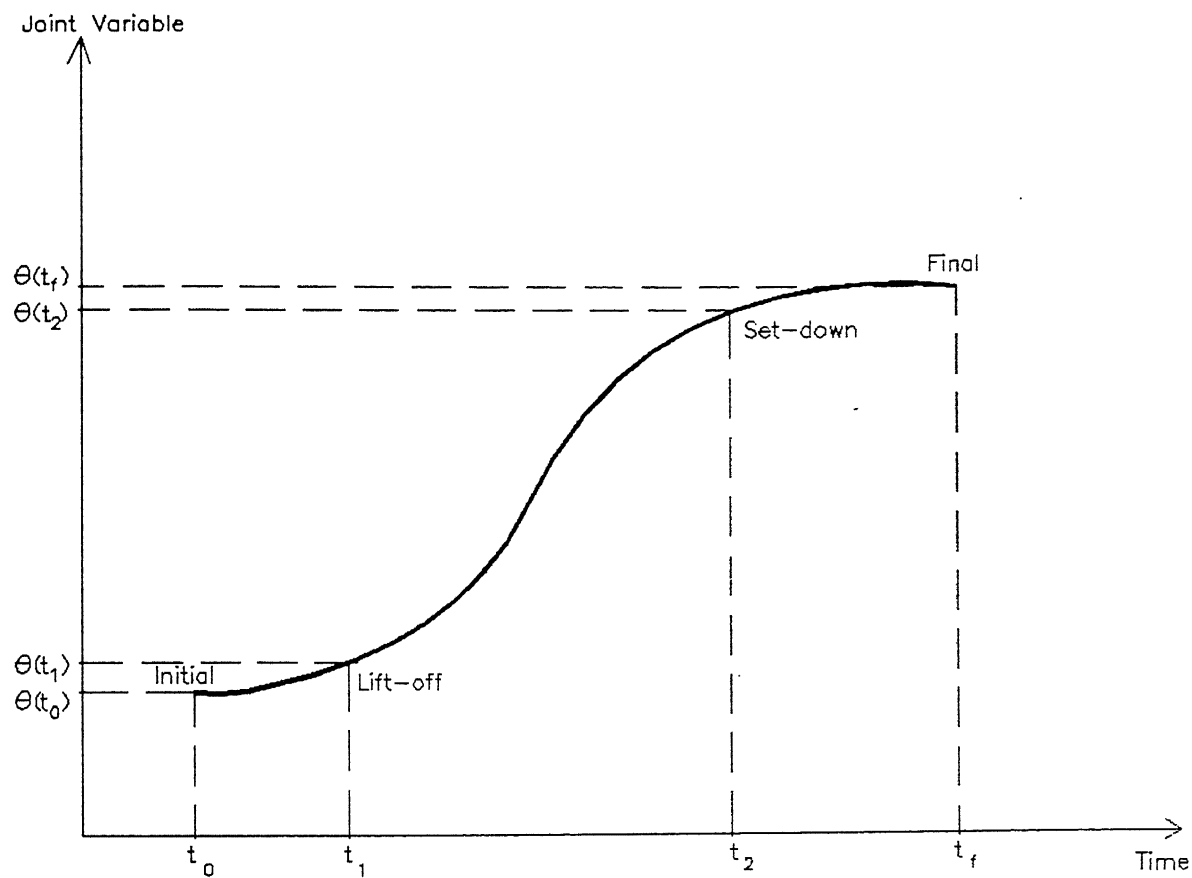


Figure 1.2 Position Condition for a Joint Trajectory

are normally zero.

(b) Lift-off position: continuous motion for intermediate points.

(c) Set-down position: same as lift-off position.

(d) Final position: same as initial position.

(vi) In addition to these constraints the extrema of all the joint trajectories must be within the physical and geometric limits of each joint.

(vii) Initial and final trajectory segment times are based on the rate of approach of the hand to and from the surface and is some fixed constant based on the characteristics of the joint motor. The mid trajectory segment time is based on the maximum velocity and acceleration limits of the joints. The maximum of all the joint times is used to plan the trajectory in a synchronous manner.

Based on the above points , a class of polynomial functions has to be selected such that the required joint position, velocity and acceleration constraints are satisfied, and the continuity of these variables is maintained over the entire time interval $[t_0, t_f]$. One approach to solve this problem is to specify a seventh-degree polynomial for each joint i .

$$q_i(t) = a_7 t^7 + a_6 t^6 + a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 \quad (1.1)$$

where the unknown coefficients a_j can be determined from the known

position and continuity conditions. However the use of such a high degree polynomial to interpolate the given knot points may not be satisfactory. An alternative approach is to split the entire joint trajectory into several segments, so that different interpolating polynomials of lower degree can be used to interpolate in each trajectory segment. There are different ways a joint trajectory can be split. The more common methods used are described below in brief.

4-3-4 Trajectory. Each joint has the following three trajectory segments: the first segment is a fourth-degree polynomial specifying the trajectory from the initial position to the lift-off position. The trajectory second is a third-degree polynomial specifying the trajectory from the lift-off position to the set-down position. The last segment is again a fourth-degree polynomial specifying the trajectory from the set-down position to final the position. The coefficients of these polynomials can be determined from the known position and continuity conditions.

3-5-3 Trajectory. This is same as the 4-3-4 trajectory, but uses polynomials of different degrees for each segment: a third-degree polynomial for the first segment, a fifth-degree for the second segment and a third-degree polynomial for the last segment.

5-Cubic Trajectory. The entire trajectory is divided into five segments, each being a cubic spline function of third-degree polynomial.

Trajectory Planning in Cartesian Coordinates

Many a times it becomes necessary to plan the trajectory in cartesian coordinates. For eg. if a straight line path is required to be traced or if there are any obstacles in the work environment to be avoided, then it is not possible to specify these constraints in joint space. Two methods to plan the trajectory in cartesian space are described in brief.

Incremental Interpolator If the path r is given then small intervals Δr_i are computed and then Δq_i is calculated as $\Delta q_i = [J]^{-1} \Delta r_i$. In this method the errors get accumulated depending on the size of Δr_i chosen. These errors have to be tracked and fed back to the controller. In this method we have to take care at singularities and near singularities, since the inverse of the Jacobian $[J]$ doesn't exist at singularities.

Absolute Interpolator The motion from a position A to another position B is expressed in terms of a "drive" transform, $D(t')$, which is a function of a normalized time t' as

$$T_o(t') = C_B(t') P_{AB} D(t') ({}^{tool}_B T_B)^{-1} \quad (1.2)$$

where

$$t' = \frac{t}{T}; \quad t' \in [0,1]$$

t is the real time since the beginning of the motion.

T is the total time for the traversal of the segment.

At position A the real time is zero, t' is zero and $D(t')$ is a 4×4 identity matrix and

$$P_{BB} = P_{AB} D(1)$$

which gives

$$D(1) = (P_{AB})^{-1} P_{AB} \quad (1.3)$$

(Details are given by Paul [1979])

1.2 Design of Planar and Spatial Curves

A variety of techniques are available for geometric design of curves. A brief review of these techniques is presented in this section for both planar and spatial curves. A curve is two-dimensional if it lies entirely in a single plane.

A curve may be represented as a collection of points or analytically by an equation or a set of equations. (Rogers and Adams) An analytical representation has several advantages. Some of these are precision, compact storage, ease of calculation of intermediate points and the properties of the curve such as slope and radius of curvature.

Analytical representation of curves originally defined by points with some boundary conditions are frequently required. From a mathematical point of view, the problem of analytically defining a curve from a known set of data points is one of interpolation. A curve that passes through all known data points is said to fit the data. One common curve fitting technique is piece-wise polynomial approximation. This technique involves determining coefficients of a polynomial of some degree. The actual shape between the points depends on the degree of polynomial and the boundary conditions imposed on it. This technique is also used to plan the trajectories in joint space as

discussed in the previous section.

Mathematically, either a parametric or non-parametric form is used to represent a curve. A non-parametric representation is either explicit or implicit. For a three dimensional space curve to be represented in its explicit non-parametric form a set of three equations

$$x = x \quad (1.4a)$$

$$y = f(x) \quad (1.4b)$$

$$z = g(x) \quad (1.4c)$$

are required, but for a planar curve only $y = f(x)$ is sufficient. In this form for each x -value only one y -value and z -value are obtained. Consequently closed or multiple-value curves cannot be represented explicitly. Implicit representations take the form $f(x,y,z) = 0$ and $g(x,y,z) = 0$ for a three dimensional curve and simply $f(x,y) = 0$ for a two dimensional case, and do not have the limitation of representing multiple-value or closed curves.

Both explicit and implicit non-parametric curve representations are axis-dependent. Thus, the choice of coordinate system affects the ease of use. For example, if in the chosen coordinate system, an infinite slope is required as a boundary condition, it cannot be directly used as a numerical boundary condition.

In parametric form of representation, each coordinate of a point on a curve is represented as a function of a single parameter. The position vector of a point on the curve is fixed by the value of the parameter. A parametric curve is expressed as

$$x = x(t) \quad (1.5a)$$

$$y = y(t) \quad (1.5b)$$

$$z = z(t) \quad (1.5c)$$

where the parameter t varies over a given range $t_1 \leq t \leq t_2$. For a two dimensional curve $x = x(t)$ and $y = y(t)$ are sufficient.

The position vector of a point on the curve is then $P(t) = [x(t), y(t), z(t)]$ for 3-D and $P(t) = [x(t), y(t)]$ for a 2-D curve. The parametric form is also suitable for representing closed and multiple-valued curves. The derivative on a 2-D curve in parametric form is represented by $P'(t) = [x'(t), y'(t)]$ where $'$ denotes the differentiation with respect to the parameter. The slope of the curve is

$$\frac{dy}{dx} = \frac{dy/dt}{dx/dt} = \frac{y'(t)}{x'(t)} \quad (1.6)$$

Study of literature shows that a wide variety of curve designing techniques are available using parametric form of representation. (Rogers and Adams, 1989).

In the intrinsic form, a three-dimensional curve in general requires two equations $\kappa = \kappa(s)$ and $\tau = \tau(s)$ for its complete description, where κ is the curvature, τ is the torsion and s is the arc length. The variables in this form are the intrinsic properties of the curve, which carry some physical meaning associated with them. For a two-dimensional curve, a well defined normal exists. However, for a three-dimensional curve any vector perpendicular to the tangent vector t is a normal vector. If n denotes the principal normal then the direction is fixed by

$$\frac{dt}{ds} = \kappa n \quad (1.7)$$

where κ is the curvature of the curve. The vector product $t \otimes n$ defines a third unit vector perpendicular to t and n known as the binormal b . In all further discussions the word normal will be used for principal normal.

The planes through a given point on the curve which contain the vectors t and n , n and b , and b and t respectively are known as the osculating plane, the normal plane and the rectifying plane (Fig 1.3). In case of a planar curve, the osculating plane is the plane of curve and the binormal vector b is fixed thus $\frac{db}{ds} = 0$ for a plane curve. On the other hand, when the curve is not a planar one, the vector b is no longer constant and an understanding of the twisted nature of the curve requires the

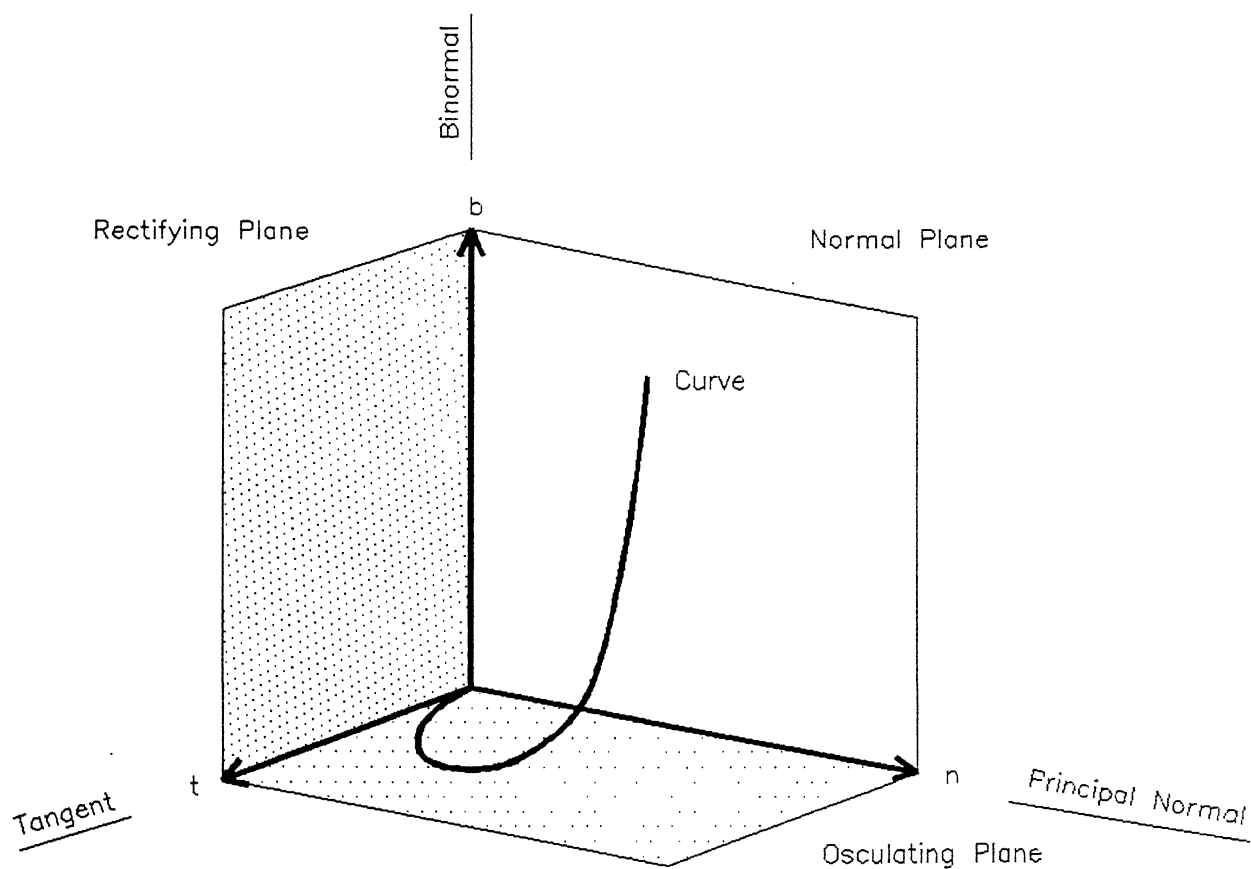


Figure 1.3 Trihedron Showing Tangent, Normal and Binormal of a Three-dimensional Curve

evaluation of $\frac{db}{ds}$ and is given by

$$\frac{db}{ds} = -\tau n \quad (1.8)$$

Curvature k gives the idea of the rate of change of slope of the curve, in other words the rate at which the curve deviates from the tangent at any point. Torsion τ is that property of the curve which gives a third dimension to the curve in cartesian space, therefore $\tau = 0$ for a planar curve. Arc length s is the length of the arc measured along the arc from a reference point. It should be remembered that the variation of these intrinsic properties k and τ as a function of arc length s gives flexibility to the design of curves.

1.3 Literature Review

Amongst the different branches of mathematics geometry is probably the most appealing to our intuition. Different branches of mathematics have evolved during the past few centuries. Differential geometry is one of them. It deals with the geometry of curves and surfaces studied by means of differential calculus. Serret-Frenet equations evolve from this branch of mathematics which form the basis of the present work.

The works of Kanayama and Miyake [6], Kanayama and Hartman [7], and Shahriar [14] form an interesting section of the literature reviewed. The concepts of shape synthesis has been developed by

Nutbourne. Shariar has solved the problem of mapping linear curvature elements in the k -s plane into the cartesian space, using the Serret-Frenet equations.

Kanayama and Miyake [6] in their work had mentioned about the trajectory specification/generation, and solves the problem using clothoid curves, and proposes a real time vehicle command system. In their work the algorithm is not described and also the velocities and acceleration have not been worked out. The trajectory generation is in two-dimensional space only.

Kanayama and Hartman [7] have taken up the problem of local path planning for autonomous vehicles. Two cost functions are defined; path curvature and derivative of path curvature. In this paper they use cubic spirals to solve the path planning problem. The resulting solutions are smoother than those obtained by using clothoid curves, but the computational complexity is comparatively higher.

Hsia and Young [5] derive analytical expressions for curvature, torsion, and radius of osculating sphere of the point trajectory in three-dimensional kinematics. Corresponding to a design position they obtain the loci of points tracing straight lines, helical curves and spherical curves.

Ryuth and Pennock [12] in this work propose an accurate method

of planning robot end effector trajectory. This method is based on the theory of ruled surface generated by a line fixed in the end effector, referred to as the tool line. The orientation of the end effector about the line is included in the analysis to completely describe the six degree-of-freedom motion of the end effector. Finally the linear and angular properties of the motion of the end effector are determined from the differential properties of the ruled surface, and utilized in the trajectory planning.

Ryuth and Pennock [13] in this work discuss the case where the ruled surface cannot be described by an explicit analytical function, then the ruled surface is expressed in terms of a single parameter. A curve generating technique is used to represent the ruled surface. The technique used in this paper is the Fergusson curve model, which generates trajectory passing through all the set points and maintaining the curvature continuity at each set point.

1.4 Objectives and Scope of Present Work

In the present work, a methodology of design for two-dimensional and three-dimensional curves, using the concept of Intrinsic geometry has been discussed. The objective of the problem is to manipulate the shape of the curve, using the shape design variables, satisfying the end position and tangent vector conditions, which are specified as input conditions.

The two types of shape models (S-C-S and C-S-C) proposed in the present work for two-dimensional shape design ensure C^2 continuity. The models proposed in the intrinsic space are mapped into the cartesian space using the Serret-Frenet equations, which have been solved numerically. The approach proposed here uses linearly varying, continuous curvature elements.

For three-dimensional curve design an alternative to Shahriar's pseudo-intrinsic approach (Shahriar, 1991) which uses cubically blended curve for modeling the rise of the helix has been attempted. In this clothoid blending (C-S-C shape model for 2-D curve design) is used to model the rise of helix. The algorithm for three-dimensional curve design is similar to that presented in Shahriar's Dissertation [14] with some modifications.

A potential application of the present work is in planning of manipulator trajectories and path planning of mobile robots (AGVs). It is worthwhile mentioning that the intrinsic properties can be directly associated with the parameters to be controlled (velocities and accelerations) while planning the trajectories. In the present work these parameters can be controlled directly and hence this approach can be used to plan the trajectories in cartesian space, with the advantage of being able to control the velocities and accelerations directly.

CHAPTER 2

INTRINSIC GEOMETRY AND CURVE DESIGN

2.1 Basic Concepts of Intrinsic Geometry

This section introduces the different concepts of intrinsic geometry used for the design of curves. Geometric elements are represented by either parametric or non-parametric equations. For a plane curve, an explicit non-parametric equation takes the form $y = f(x)$, an implicit non parametric equation takes the form $f(x,y) = 0$. On the other hand, for a plane curve to be represented in the parametric form, a set of two equations $x = x(u)$, $y = y(u)$, with u as the parameter ($u_1 \leq u \leq u_2$) are necessary.

In the intrinsic form, a three-dimensional curve in general requires two equations $\kappa = \kappa(s)$ and $\tau = \tau(s)$ for its complete description, where κ is the curvature, τ is the torsion and s is the arc length. As opposed to the intrinsic form, neither of the above forms have the ability to control the intrinsic properties of a curve explicitly. If the intrinsic properties of a curve can be associated with some useful variables to be controlled, then the intrinsic shape synthesis becomes more attractive.

For the case of a 2-D curve, $\tau = 0$. In this case, the second order Serret - Frenet equations can be solved for any given

function of $k(s)$. For the purpose of trajectory planning piecewise linear curvature functions are sufficient as explained in Chapter 1. Consequently ramp type continuous curvature functions for $k(s)$ have been considered.

2.1.1 Serret - Frenet equations

Consider a Curve C as shown in Figure 2.1. Let r be the position vector of a generic point P and let s the arc length of P from a reference point A , be the parameter describing the curve as $r(s)$. The unit tangent vector, the curvature, the torsion, the normal, the binormal of the curve C at the point P are given as follows.

$$t = \frac{dr}{ds} \quad (2.1)$$

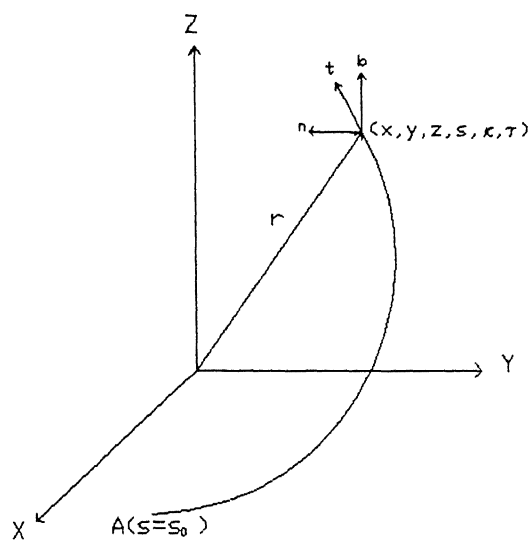
$$\frac{dt}{ds} = k n \quad (2.2)$$

$$\frac{dn}{ds} = -k t + \tau b \quad (2.3)$$

$$\frac{db}{ds} = -\tau n \quad (2.4)$$

$$b = t \otimes n \quad (2.5)$$

where t is the unit tangent vector, n the unit normal vector, b is the unit binormal vector, k is the curvature, and τ is the torsion. Equations (2.2), (2.3) and (2.4) are known as the



Cartesian Space of the Curve

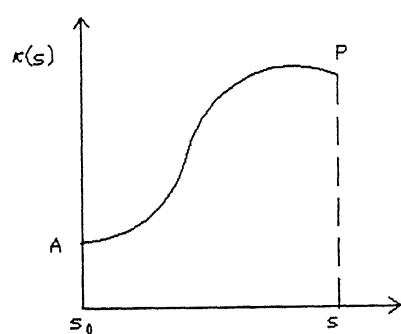
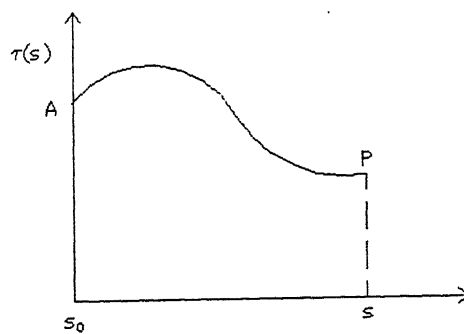
 $\kappa - s$ Plane $\tau - s$ Plane

Figure 2.1 Intrinsic Geometry of a Three-dimensional Curve

formulae of Serret-Frenet (Faux and Pratt, 1980). These equations are used as the basis of shape synthesis for the present work.

If a curve is planar then $\tau = 0$ and the above equations can be written down as follows.

$$\mathbf{r} = \begin{bmatrix} x(s) \\ y(s) \end{bmatrix} \quad (2.6)$$

$$\mathbf{t} = \frac{d\mathbf{r}}{ds} \quad \text{and} \quad \mathbf{n} \cdot \mathbf{t} = 0 \quad (2.7)$$

$$\frac{d\mathbf{t}}{ds} = \kappa \mathbf{n} \quad (2.8)$$

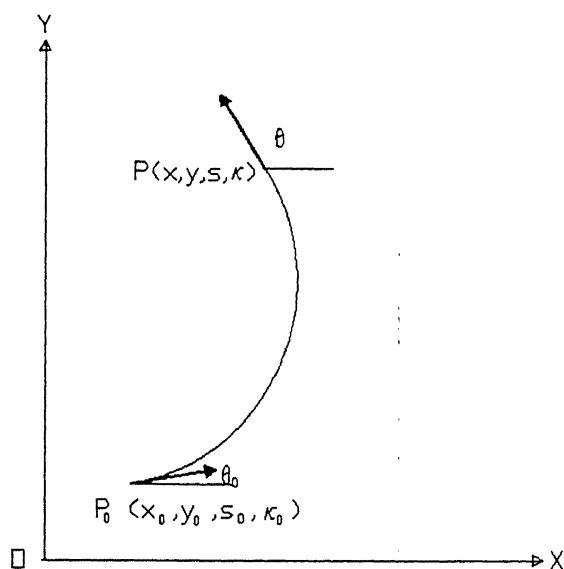
$$\frac{d\mathbf{n}}{ds} = -\kappa \mathbf{t} \quad (2.9)$$

The review of literature on 2-D and 3-D curves indicate that different mathematical strategies have been used to find the general solution of these curves using the above equations. The approach used in this study is discussed in the following section.

2.1.2 Solution Approach for Two Dimensional Problems

Let us consider the problem of defining a curve starting with a point $P_0(x_0, y_0)$ in a two dimensional space (Fig. 2.2). Let us also assume that the tangent directions at P_0 has been specified as θ_0 , and the arc length from a reference point A as s_0 . If κ is the curvature at any point P, then it is assumed that the variation of κ as a function of arc length as the parameter, $\kappa =$

$k(s)$, has been specified. It can be seen that $k(s)$ defines the



Cartesian Space of a Two-dimensional Curve

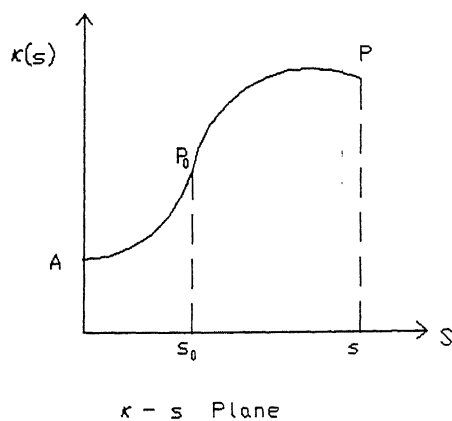


Figure 2.2 Two-dimensional Curve Design Using $\kappa - s$ Plane

shape of the curve. The problem of finding the cartesian coordinates x and y as functions of the arc length s as parameter can be solved by using Serret-Frenet equations (2.2) through (2.4). It should be remembered that the variation of intrinsic property curvature κ gives flexibility to the shape synthesis (Shahriar, 1991).

To begin with let us rewrite the equation as follows:

$$t = \frac{dr}{ds} = \begin{bmatrix} \frac{dx(s)}{ds} \\ \frac{dy(s)}{ds} \end{bmatrix} \quad (2.10)$$

Since $n \cdot t = 0$, then

$$n = \begin{bmatrix} -\frac{dy(s)}{ds} \\ \frac{dx(s)}{ds} \end{bmatrix} \quad (2.11)$$

On the other hand :

$$\frac{dt}{ds} = \begin{bmatrix} \frac{d^2x(s)}{ds^2} \\ \frac{d^2y(s)}{ds^2} \end{bmatrix} \quad (2.12)$$

Substituting equations (2.11) and (2.12) in equation (2.2) yields:

$$\begin{bmatrix} \frac{d^2 x(s)}{ds^2} \\ \frac{d^2 y(s)}{ds^2} \end{bmatrix} = k(s) \begin{bmatrix} -\frac{dy(s)}{ds} \\ \frac{dx(s)}{ds} \end{bmatrix} \quad (2.13)$$

Therefore,

$$\frac{d^2 x}{ds^2} + k(s) \frac{dy}{ds} = 0 \quad (2.14)$$

$$\frac{d^2 y}{ds^2} - k(s) \frac{dx}{ds} = 0 \quad (2.15)$$

where at $s = s_0$, $x = x_0$, $y = y_0$, $\theta = \theta_0$.

Let us assume that $k(s)$ is defined as a piecewise linear function of s . Let

$$\eta' = x' + iy' \quad (2.16)$$

where

$$\eta' = \frac{d\eta}{ds}, \quad x' = \frac{dx}{ds}, \quad \text{and} \quad y' = \frac{dy}{ds}$$

The governing equations (2.14) and (2.15) can now be written as follows.

$$(\eta')' - i k(s) (\eta') = 0 \quad (2.17)$$

or

$$\eta' = e^{i \int k(s) ds} = e^{i\theta(s)} \quad (2.18)$$

Separating the real and imaginary parts, we get

$$\frac{dx}{ds} = \cos[\theta(s)] \quad (2.19)$$

$$\frac{dy}{ds} = \sin[\theta(s)] \quad (2.20)$$

Integrating these equations with respect to the arc length s yields the parametric coordinates of the 2-D curve in terms of the intrinsic parameter s .

$$x(s) = \int_{s_0}^s \cos[\theta(\sigma)] d\sigma + x_0 \quad (2.21)$$

$$y(s) = \int_{s_0}^s \sin[\theta(\sigma)] d\sigma + y_0 \quad (2.22)$$

where

$$\theta(\sigma) = \int_{s_0}^{\sigma} k(s) ds + \theta_0 \quad (2.23)$$

Equations (2.21) through (2.23) can be solved by using numerical integration and for the cartesian coordinates $x(s)$ and $y(s)$ once the curvature function $k(s)$ is known.

2.2 Planar Curve Design:

This section deals with the design of planar curves for linear curvature segments. Let us consider the problem of defining a curve passing through two points $P_0(x_0, y_0)$ and $P_1(x_1, y_1)$ in a plane, the tangent angles at P_0 and P_1 being

specified as θ_0 and θ_1 .

The above problem of designing a 2-D curve passing through two given points with their end tangents specified can be divided in the following steps

- (i). For k as the curvature at any point P , the variation of k as a function of arc length is defined. This step involves shape model definition as it decides the geometry of the curve uniquely.
- (ii). Shape model being defined, equations (2.21), (2.22) and (2.23) are solved simultaneously for the complete definition of the curve in cartesian space.

2.2.1 Shape Models

In this section the shape models considered for this study are described below.

Straight line Clothoid Straight line (S-C-S) Model: In S-C-S model the two end portion of the curve are straight lines and they are blended with a clothoid pair (symmetric cornu spirals) in between. The piecewise linear and continuous curvature will be either of the following two types as in figure (2.3) and (2.4).

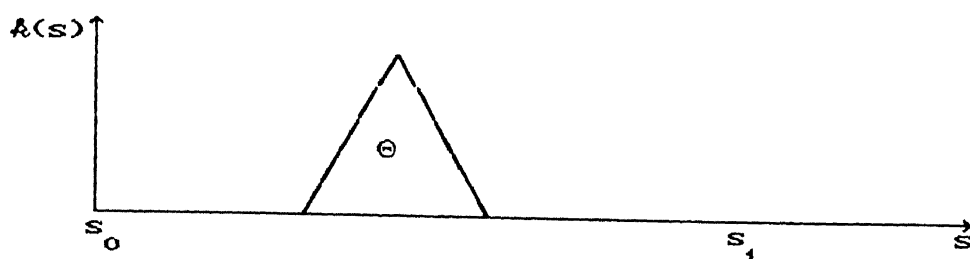


Figure 2.3: S-C-S Shape Model with Positive change in tangent angle.

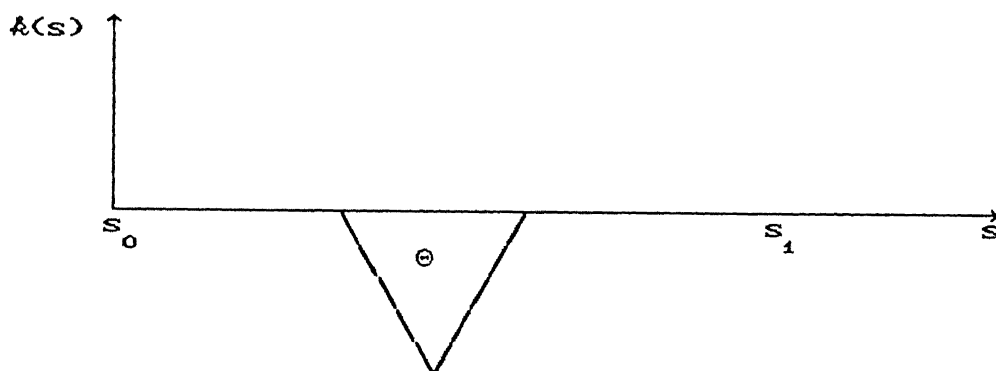


Figure 2.4: S-C-S Shape Model with Negative change in tangent angle.

Area of the triangle Θ represents the change in the tangent angle, therefore a triangle below the s -axis indicates a negative change in the tangent angle. Since we assume a symmetric nature of the curve at the blend the shape model uniquely defines the curve in cartesian space.

Clothoid Straight line Clothoid (C-S-C) Model: In C-S-C model the two end portions of the curve are pairs of symmetric spirals and the middle portion is a straight line. The piecewisely linear curvature function is either one of the following four types as in

figures (2.5) to (2.8). The areas of the triangles Θ_1 and Θ_2 represents the change in tangent angle of the first and second Clothoid pairs respectively.

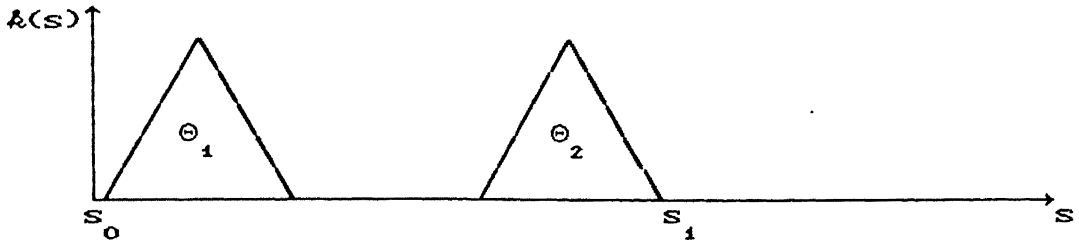


Figure 2.5: C-S-C Shape Model with positive tangent angle change for both the Clothoid pairs.

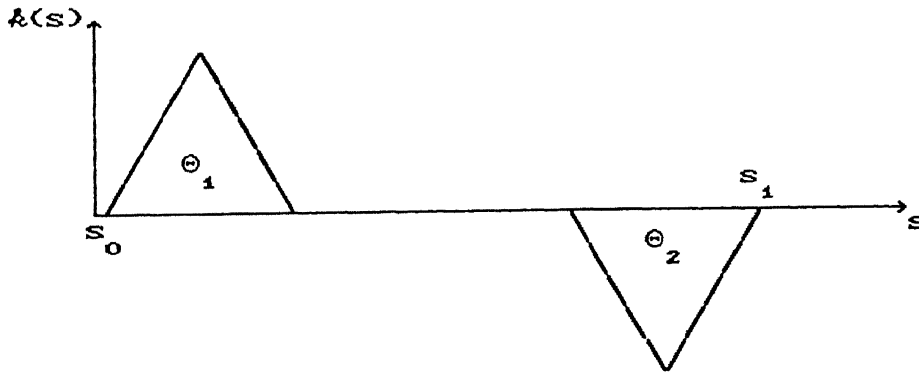


Figure 2.6: C-S-C Shape Model with positive tangent angle change for the first Clothoid pair and Negative for the second.

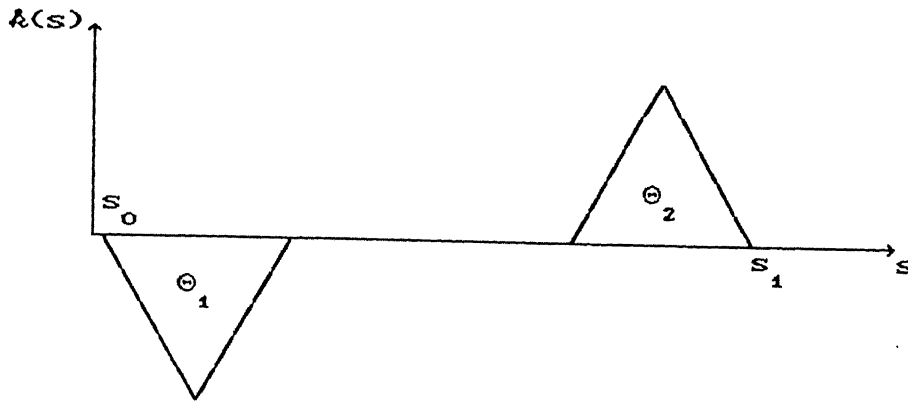


Figure 2.7: C-S-C Shape Model with negative tangent angle change for the first Clothoid pair and positive for the second.

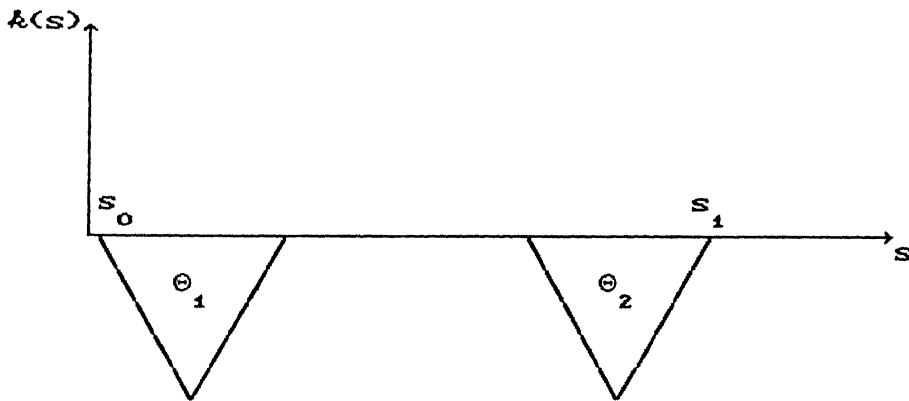


Figure 2.8: C-S-C Shape Model with negative tangent angle change for both the Clothoid pairs.

A triangle to the bottom indicates a negative change in the tangent angle. In both the shape models the slope of the straight

lines in the k - s plot is called the sharpness λ .

Design Methodology for S-C-S Model:

The input parameters are the position coordinates of the end points $P_0(x_0, y_0)$, $P_1(x_1, y_1)$, the tangent angles θ_0 , θ_1 , and λ_d the desired sharpness.

The first step is to find the point of intersection $Q(x_4, y_4)$ of the two tangents using the following expressions (Figure 2.9).

$$x_4 = \frac{(m_0 x_0 - m_1 x_1) + (y_1 - y_0)}{(m_0 - m_1)} \quad (2.24)$$

and

$$y_4 = m_0(x_4 - x_0) + y_0 \quad (2.25)$$

where

$m_0 = \tan \theta_0$ and $m_1 = \tan \theta_1$, the slopes of the initial and final tangents.

Let λ_d be the given sharpness, then over the arc length of the clothoid pair the variation of curvature and the sharpness will be as shown in figures (2.10) and (2.11).

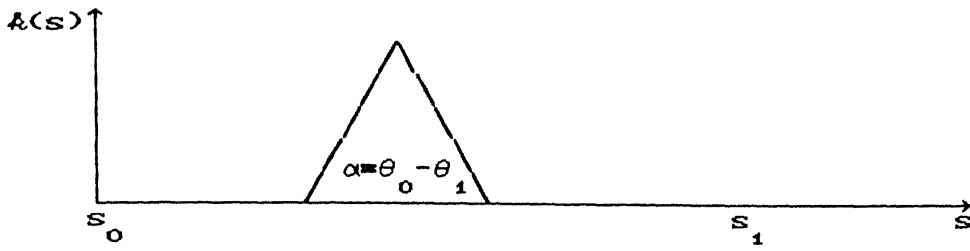


Figure 2.10 : S-C-S Shape Model with Positive change in tangent angle.

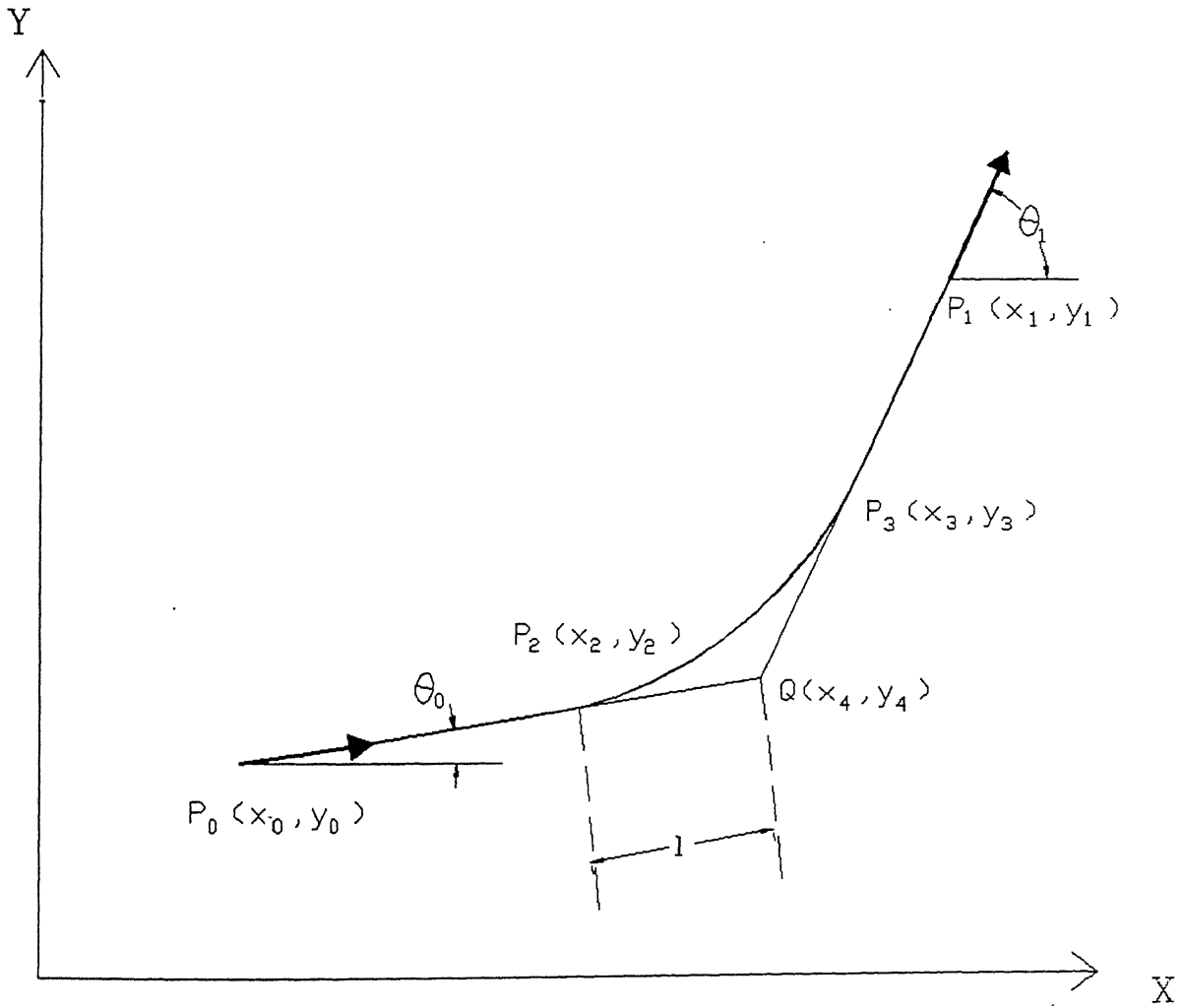


Figure 2.9 Cartesian Space of a S-C-S Model Planar curve

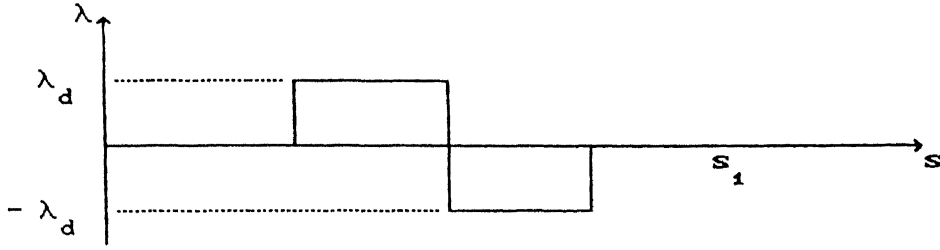


Figure 2.11: Variation of sharpness Vs. arc length for S-C-S model

If s' is the total arc length of the clothoid pair which blends the two tangents and $\alpha = \theta_1 - \theta_0$ the area under the k - s curve $= \alpha$.

The area α expressed in terms of λ_d and s' from the geometry of k - s plot is

$$\alpha = \frac{\lambda_d s'}{4} \quad (2.25)$$

or

$$s' = 2 \sqrt{\frac{\theta_1 - \theta_0}{\lambda_d}} \quad (2.26)$$

The expressions for the cartesian coordinates as derived earlier (equations 2.21, 2.22 and 2.23) are

$$x(s) = \int_{s_0}^s \cos[\theta(\sigma)] d\sigma + x_0$$

$$y(s) = \int_{s_0}^s \sin[\theta(\sigma)] d\sigma + y_0$$

$$\theta(\sigma) = \int_{s_0}^{\sigma} k(s) ds + \theta_0$$

Since $k(s)$ is a linear function of s , we substitute $k_0 + \lambda_d s$ for $k(s)$ in the above equations. The equations for the cartesian coordinates in terms of λ_d , k_0 , θ_0 and t a dummy variable will be

$$x = x_0 + \int_0^s \cos \left[\lambda_d t^2 / 2 + k_0 t + \theta_0 \right] dt \quad (2.27)$$

$$y = y_0 + \int_0^s \sin \left[\lambda_d t^2 / 2 + k_0 t + \theta_0 \right] dt \quad (2.28)$$

Since we also assume that the curve starts with zero curvature $k_0 = 0$, and using suitable transformations for the proper choice of coordinate system (i.e.) rotating the axis so that the x -axis is coincident with the initial tangent we can write the following equations.

$$x_3 - x_2 = \int_0^{s'} \cos(\lambda_d t^2 / 2) dt \quad (2.29)$$

$$y_3 - y_2 = \int_0^{s'} \sin(\lambda_d t^2 / 2) dt \quad (2.30)$$

Let $x_3 - x_2$ be 'a' and $y_3 - y_2$ be equal to 'b'

From the geometry (Figure 2.9) $l(\cos\theta_1 + \cos\theta_2) = x_3 -$

$$x_2 = a$$

Therefore

$$l = \frac{a}{(\cos\theta_1 + \cos\theta_2)} \quad (2.31)$$

where l is the exit distance of the clothoid pair.

Now $P(x_2, y_2)$ can be calculated from the following expressions.

$$\left. \begin{aligned} x_2 &= x_4 - l \cos(\theta_1) \\ y_2 &= y_4 - l \sin(\theta_1) \end{aligned} \right\} \quad (2.32)$$

where l is given by eqn (2.31) and $Q(x_4, y_4)$ by eqn (2.24) and eqn (2.25)

Calculating all the unknowns the points on the curve at regular intervals of the parameter arc length s are calculated.

Design Methodology for C-S-C Model

The input parameters again in this case are the position coordinates of the end points $P_0(x_0, y_0)$, $P_1(x_1, y_1)$, the tangent angles θ_0 , θ_1 , and λ_d the desired sharpness.

Let θ_m be the tangent angle made by the intermediate straight line portion of the curve as shown in Figure 2.12. In this model one more assumption is made that the sharpness of the clothoid pairs at both the ends is same and also that they are symmetric within themselves. The following figure shows the cartesian output of a C-S-C shape model.

From the same arguments done for the S-C-S model in the previous paragraphs

$$\theta_m - \theta_0 = \lambda_d \frac{s_2^2}{4} \quad \text{or}$$

$$s_2 = 2 \sqrt{\frac{\theta_m - \theta_0}{\lambda_d}} \quad (2.33)$$

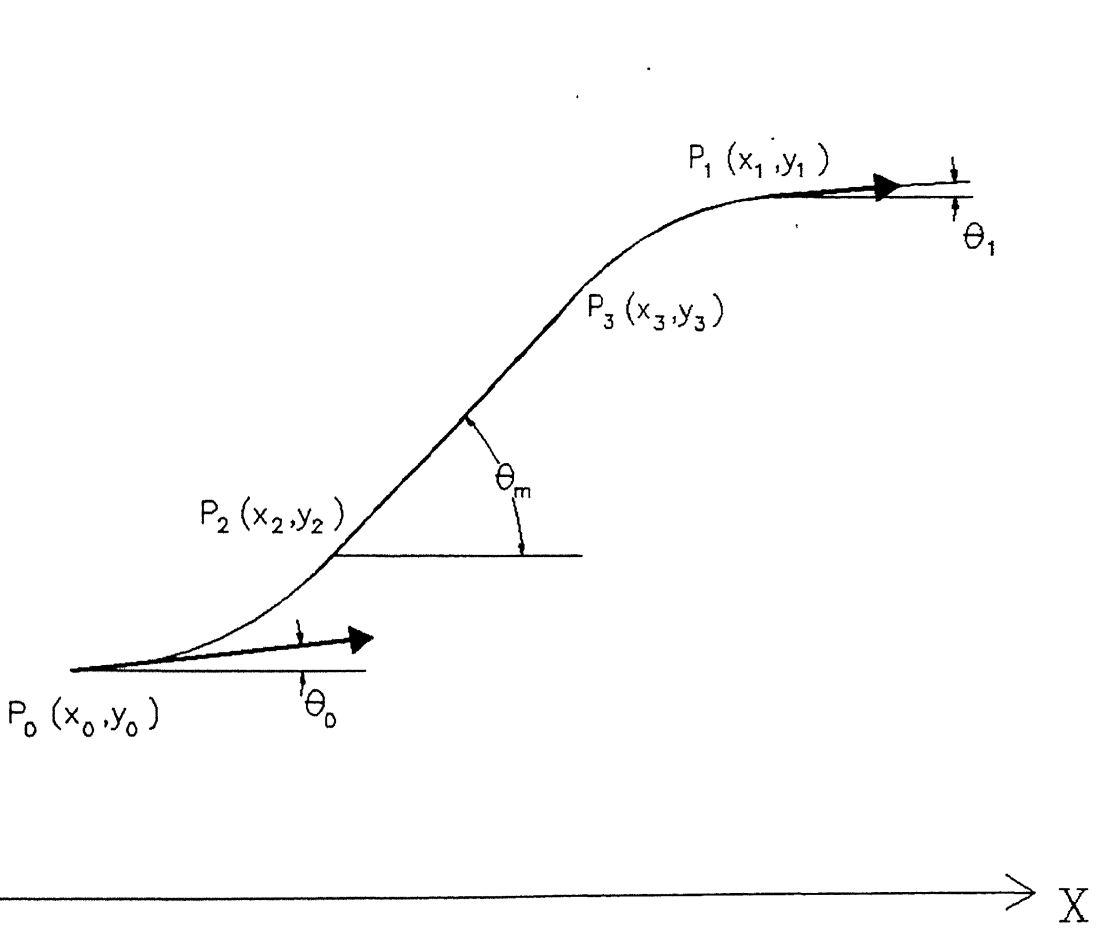


Figure 2.12 Cartesian Space of a C-S-C Shape Model Planar Curve

By suitable coordinate transformations (i.e) rotating the axes system such that the x-axis coincides with the initial tangent, θ_0 becomes equal to zero

$$s_2 = 2 \sqrt{\frac{\theta_m}{\lambda_d}} \quad (2.34)$$

similarly for the second clothoid pair also

$$\theta_1 - \theta_m = \frac{\lambda_d (s_1 - s_3)^2}{4} \quad \text{or}$$

$$s_3 = s_1 - 2 \sqrt{\frac{\theta_m - \theta_0}{\lambda_d}} \quad (2.35)$$

To compute the values of s_2 and s_3 it is necessary to know the value of θ_m before hand. This is computed iteratively as follows.

- (i) Since there is no basis to guess the value of θ_m , the initial guess made is the one equal to the angle made by the line joining the initial and final points.
- (ii) At this stage knowing the initial guess value of θ_m , s_2 and s_3 are computed based on this value of θ_m . Subsequently $P_2(x_2, y_2)$ and $P_3(x_3, y_3)$ are also computed.
- (iii) The value of θ_m will be updated to the value equal to the angle made by the line joining $P_2(x_2, y_2)$ and $P_3(x_3, y_3)$.
- (iv) Here the old value of θ_m is compared with the updated value of θ_m . If the difference is less than the acceptable limit then

the new value of θ_m is finalized and is used for all further calculations.

(v) If the difference is greater than the acceptable value then repeat step numbers (ii), (iii) and (iv).

The initial guess for θ_m can also be initialized to a constant value, but if it differs largely from the actual value then it takes a large time to converge. With the initial guess as described above the algorithm converges quite faster. This initial guess also confirms for the shortest path only, even if any another solution exists. Once the value of θ_m is calculated, the points on the curve are computed as follows.

The values of s_2 and s_3 are computed from the expressions (2.34) and (2.35). Subsequently by using the expressions (2.27) and (2.28) for cartesian coordinates the points on the curve are computed at regular and small intervals of the arc length.

2.3 Planar Trajectory Design

Trajectory planning involves the definition of time derivatives in addition to the definition of position vector itself. The problem will be completely solved when the expressions for r , $\frac{dr}{dt}$, $\frac{d^2r}{dt^2}$, are known completely as a function of time, when either s or $\frac{ds}{dt}$ is given as a function time.

The expressions for velocity and acceleration can be written as follows.

$$\frac{dr}{dt} = \frac{dr}{ds} \frac{ds}{dt} \quad (2.36)$$

$$\frac{d^2r}{dt^2} = \frac{d^2r}{ds^2} \left(\frac{ds}{dt} \right)^2 + \frac{dr}{ds} \frac{d^2s}{dt^2} \quad (2.37)$$

To calculate the values of velocities and accelerations from the above expressions it is necessary to know $\frac{ds}{dt}$ and $\frac{d^2s}{dt^2}$ apriori. The variation of $\frac{dr}{ds}$ and $\frac{d^2r}{ds^2}$ as a function of arc length is also required. For $\frac{dr}{ds}$ and $\frac{d^2r}{ds^2}$ the following expressions are used.

$$x = \int_{s_0}^s \cos[\theta(s)] ds + x_0 \quad (2.38)$$

$$y = \int_{s_0}^s \sin[\theta(s)] ds + y_0 \quad (2.39)$$

Differentiating under the integral sign

$$x' = \frac{dx}{ds} = \cos[\theta(s)] \quad (2.40)$$

$$y' = \frac{dy}{ds} = \sin[\theta(s)] \quad (2.41)$$

Again differentiating x' and y' with respect to s .

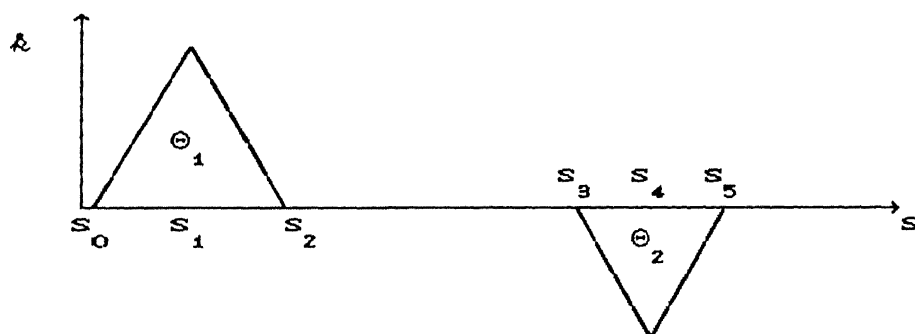
$$x'' = \frac{d^2 x}{ds^2} = -\sin\theta(s) \frac{d\theta(s)}{ds} \quad (2.42)$$

$$y'' = \frac{d^2 y}{ds^2} = \cos\theta(s) \frac{d\theta(s)}{ds} \quad (2.43)$$

Where

$$\theta(s) = \int_0^s k(s) ds + \theta_0 \quad (2.44)$$

The related expressions for $k(s)$, $\theta(s)$ and $\theta'(s)$ for the C-S-C model as in figure are given below.



For $s_0 \leq s \leq s_1$:

$$k(s) = \lambda_d s$$

$$\theta(s) = \frac{\lambda_d s^2}{2} + \theta_0, \quad \theta_0 = 0 \quad \text{and} \quad \theta' = \frac{d\theta(s)}{ds} = \lambda_d s$$

For $s_1 \leq s \leq s_2$:

$$k(s) = \lambda_d s + 2k_1$$

$$\theta(\sigma) = 2k_1\sigma - \frac{1}{2}\lambda_d\sigma^2 - 2k_1s_1$$

$$\theta' = \frac{d\theta(s)}{ds} = 2k_1 - \lambda_d s$$

For $s_2 \leq s \leq s_3$:

$$k(s) = 0$$

$$\theta(s) = k_1 s_1 \quad \text{and} \quad \theta' = \frac{d\theta(s)}{ds} = 0$$

For $s_3 \leq s \leq s_4$:

$$k(s) = -\lambda_d s + \lambda_d s_3$$

$$\theta(\sigma) = k_1 s_1 + \lambda_d (\sigma - s_3) \left\{ s_3 - \frac{(\sigma + s_3)}{2} \right\}$$

$$\theta' = \frac{d\theta(s)}{ds} = \lambda_d (s_3 - s)$$

For $s_4 \leq s \leq s_5$:

$$k(s) = \lambda_d s - \lambda_d s_5$$

$$\theta(\sigma) = k_1 s_1 + \frac{1}{2}(s_4 - s_3)k_4 + \lambda_d \left(\frac{\sigma^2}{2} - s_5 \sigma - \frac{s_4^2}{2} + s_4 s_5 \right)$$

$$\theta' = \frac{d\theta(s)}{ds} = \lambda_d (s - s_5)$$

The expressions for rest of the models can be written similarly with the required changes for the expressions of $k(s)$.

Specific numerical examples of S-C-S and C-S-C shape model

curve design of all the types discussed above are discussed in Chapter 4. One example of trajectory planning for C-S-C shape model with $ds/dt = \text{constant}$ is also considered in Chapter 4.

CHAPTER 3

SPATIAL TRAJECTORY DESIGN

3.1 Methodology of Three-dimensional Curve Design

This section deals with the design of spatial curves. The problem could be stated as: Define a space curve passing through two points A and B (Fig 3.1); the two unit end tangent vectors at A and B being given as t_A and t_B respectively.

The desired space curve is proposed to be a generalized helix having a base curve on the plane normal to the skew direction and a rise curve along the skew direction. The above stated problem can broadly be divided into the following three steps.

- (i) Design a two-dimensional curve in a plane perpendicular to the skew direction $A' A''$ (Fig 3.1) containing t_A .
- (ii) Design a 2-D curve for defining the rise of the helix (3-D curve) as a function of the arc length of the base curve designed in step i.
- (iii) Using the definitions of these two curves a 3-D curve is defined as proposed below.

Approach for 3-D Curve Design

The design of 3-D curve is not done directly in the coordinate system in which the input data is given, which will be

further referred to as the global coordinate system. A local coordinate system O_1 -UVW is constructed such that O_1 is coincident with C (Fig 3.1). C is the point of intersection of the common normal to the tangents t_A and t_B and the tangent direction t_A . U-axis is along the direction t_A , the W-axis is along the direction of the common normal $A'-A''$ and the V-axis is such that $W = U \otimes V$. Let Q_1 and Q_2 be the projections of the end points A and B on the O-UV plane, and let m_1 and m_2 be the projections of the unit tangent vectors t_A and t_B on UV plane. The angle that m_2 makes with m_1 is denoted by Θ where $\Theta = \cos^{-1}(m_1 \cdot m_2)$. At this stage it should be noted that the local coordinate system O-UVW is selected as above for the reason that the unit tangent vectors t_A and t_B are projected onto the UV plane with its full length, and there is no component of either t_A or t_B along the W-axis. This makes us more free to design the third dimension of the curve. Setting up the local coordinate system we proceed with the actual design of the curve in this coordinate system and then transfer back the points obtained in the local coordinate system to the global coordinate system.

To begin with, it is necessary to define a planar curve with specified end coordinates and tangent vectors in the UV plane. The end points for the base curve will be the projections of the points A and B on the UV plane, they are represented by Q_1 and Q_2 . Similarly the end tangents will be the projections of t_A and t_B on the UV plane, they are represented by m_1 and m_2 . With this data a

base curve is constructed using the method proposed in Chapter 2. it consists of a set of linear curvature segments which will satisfy end point geometric constraints of the UV plane.

Once the base curve is defined, it is necessary to describe the rise of the helix (along the W direction) as a function of arc length of the base curve (denoted by S'). In other words, it is now necessary to define another curve in the $W - S'$ plane. The two end points of this curve corresponding to points A and B are $(S_A, 0)$ and (S_B, d) where d is the skew distance between t_A and t_B . Without loss of generality, it can be assumed that $s_A = 0$ and $s_B = s^*$ where s^* is the total arc length of the base curve. Furthermore this rise curve tangents should be horizontal that is parallel to the S' axis.

One can think of several different ways to conceive the geometry of this rise curve. This rise curve can again be modeled using the intrinsic geometry approach. Curvature continuity of the rise curve is also required to be maintained for G^2 continuity to be maintained in the 3-D helix designed, keeping this in mind the following two methods of modeling the rise curves are described below which are a compromise between the curvature continuity and computational complexity.

- Clothoid Blended Curve (C-S-C shape model using intrinsic geometry).
- Cubically Blended Curve.

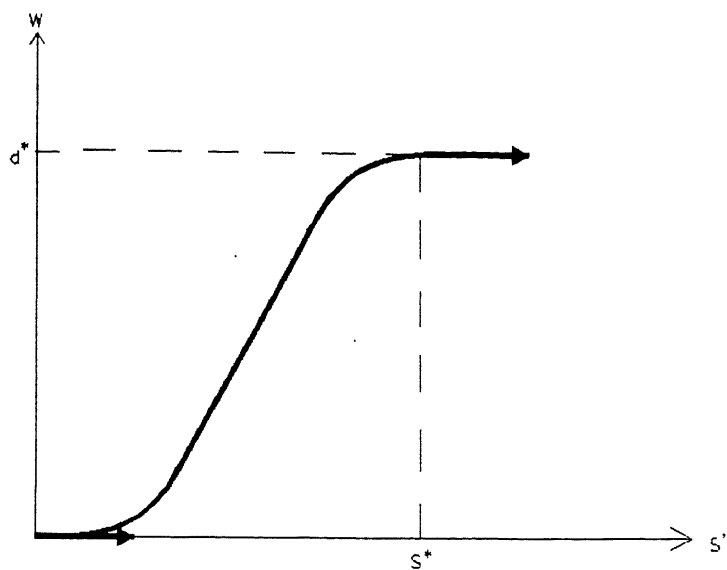


Figure 3.2 Clothoid Blended Curve with a Straight Line portion
in between to Model the Rise of a 3-D Curve

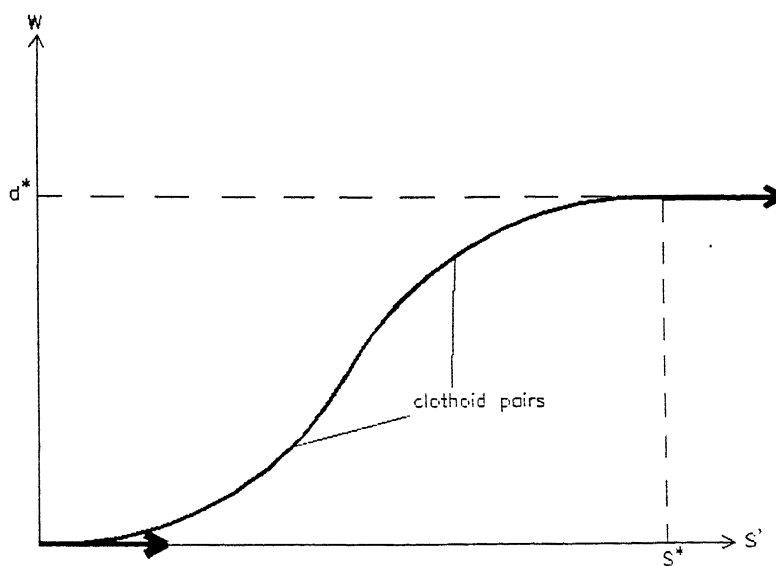


Figure 3.3 Clothoid Blended Curve without a Straight Line Portion
in between to Model the Rise of a 3-D Curve

Clothoid Blended Curve

Figures 3.2 and 3.3 show the schematic concept of a clothoid blended curve. It consists of two pairs of clothoid segments at both the ends A and B, and a linear segment in the middle if possible as in Figure 3.2. The sharpness of these two clothoid pairs is also equal to the base curve sharpness, if a straight line portion in the middle is possible, or is the minimum sharpness without a straight line portion in between (Fig 4.3). The tangent angles at both the ends being zero, in other words both the tangents are horizontal, the rise curve is uniquely defined as C-S-C model, exactly on the lines of the C-S-C model discussed in Chapter 2.

Though the cubically blended rise curve gives a very good control over the intrinsic properties of the curve, it is a computationally costly model, since the whole algorithm for C-S-C curve has to be repeated and in relating the base curve coordinate (U and V) to the rise coordinate (W coordinate) we land up with solving of non-linear equation.

Cubically Blended Curve

This model differs from the above in the respect that the two ends are blended with Cubic Curves instead of clothoid pairs as in in clothoid blended curve. A straight line portion is also kept in between invariably, with the curvature continuity being

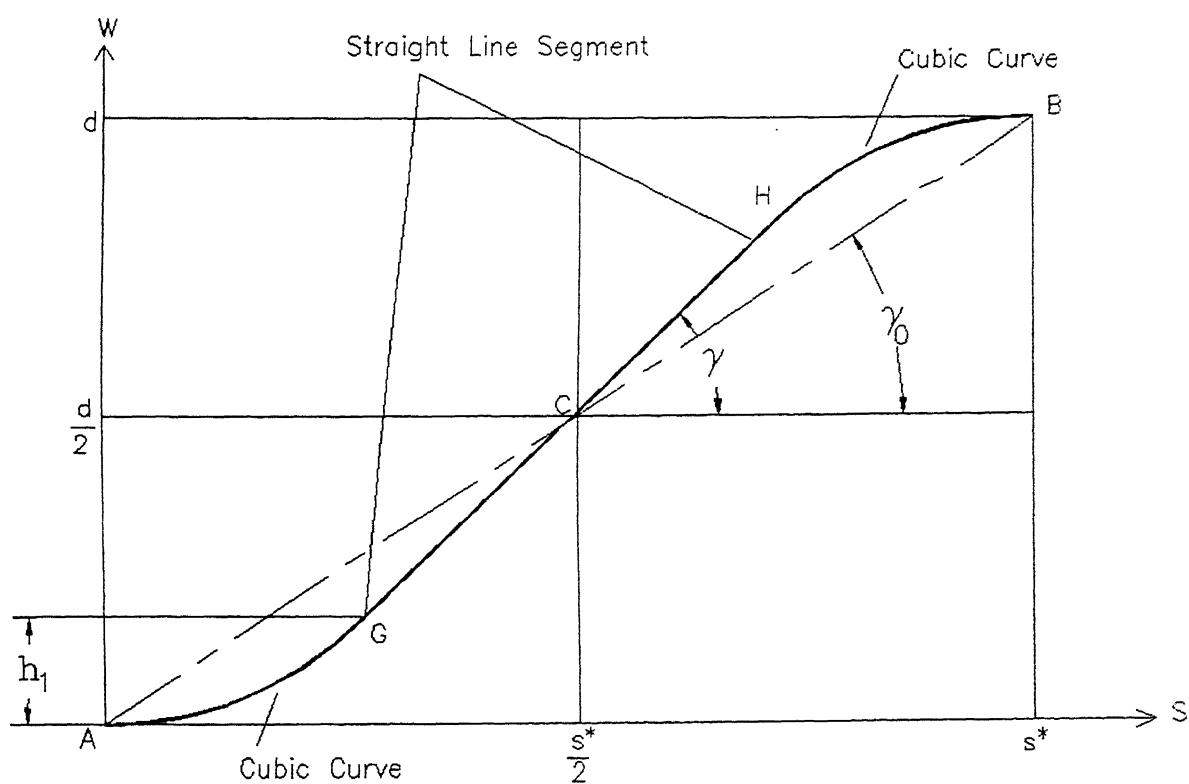


Figure 3.4 Cubically Blended Curve For Modelling the rise of Three-dimensional Curve

maintained (Figure 3.4). Consider a cubic required to be designed between points A and G. This can be accomplished as follows. Since we know the points $B(d, s^*)$, the point $C(\frac{d}{2}, \frac{s^*}{2})$ is also known. Draw a line through C with a slope of $\tan(\gamma)$.

Let the equation of the Cubic Curve be

$$r = a_0 + a_1 u + a_2 u^2 + a_3 u^3 \quad (3.1)$$

$$0 \leq u \leq 1$$

Now the cubic curve has to pass through the point A and should be tangential to the S' -axis at point A. Further, at its other end ($u=1$), the cubic curve should have zero curvature to maintain the curvature continuity with the straight line, and should be tangential to the line passing through C, having a slope $\tan(\gamma)$.

The boundary conditions can be specified as:

(i) End point condition. At $u=0$, $\frac{dr}{du} = [k_0 \quad 0]^T$

(ii) End point tangent conditions: At $u=0$, $\frac{dr}{du} = [k_0 \quad 0]^T$

(iii) End point tangent conditions: At $u = 1$,

$$\frac{dr}{du} = [k_1 \cos \gamma \quad k_1 \sin \gamma]^T$$

(iv) End point curvature condition. At $u=1$, $\frac{dr}{du} \otimes \frac{d^2 r}{du^2} = 0$

where k_0 and k_1 are constants to be evaluated along with a_0 , a_1 , a_2 and a_3 .

Let h_1 be the height where the end point G of the cubic

blends with the straight line segment. Solving the boundary condition equations for the known coefficients we get,

$$a_0 = 0$$

$$a_1 = [k_0 \quad 0]^T$$

$$a_2 = -a_1$$

$$a_3 = \left[\frac{1}{3} (k_1 \cos \gamma + k_0) \quad \frac{1}{3} (k_1 \sin \gamma) \right]^T$$

$$k_0 = 3 \left[\frac{s^*}{2} - \frac{d/2}{\tan \gamma} \right]$$

$$k_1 = \frac{3h_1}{\sin \gamma}$$

$$\gamma > \gamma_0 = \tan^{-1} \left[\frac{d}{s^*} \right]$$

Finally, the cubic curve AG is defined as:

$$r = a_1 u + a_2 u^2 + a_3 u^3$$

It is appropriate to mention at this stage that for the complete definition of the cubic curve, proper choice of values of γ and h_1 have to be made so that all the constants can be evaluated. The values $\gamma = \gamma_0 + 4^\circ$ and $h_1 = 0.25 d$ is moderately a good choice. The equation of the straight line portion is

$$y = \tan \gamma + \frac{s^*/2}{\frac{d}{2} \tan \gamma} \quad (3.4)$$

It should be noted that the curve HB can be defined in a similar way. The rise curve will now consist of four segments AG, GC, CH and HB. One can complete the points on the segments AG and GC according to the above procedure and the points on the segments CH and HB can be obtained using transformations of mirror reflections around W and S' axis. The cubically blended curve is definitely superior to the parabolically blended curve in terms of computational efficiency. It is difficult to exercise a direct control on the curvature κ of the rise curve, hence the 3-D curve design approach using cubically blended curve model for the rise curve (or the helix angle) can be called as Pseudo-intrinsic approach.

Having defined the base curve and the rise curve, the next step would be to correlate these two curves to define the desired 3-D curve. The U and V coordinates of the base are computed for small intervals of arc length, at this stage only the W coordinate is also computed for the cubic curve as follows.

The y-coordinate of the rise curve will be the W coordinate of the 3-D curve. Since we know the curve length of the base curve which is x-coordinate of the rise curve, the portion to which this point corresponds to in the rise curve can be easily found out (cubic or straight line portion). Knowing this, the value of u (if it lies in the cubic region) can be computed by Newton-Raphson

method and subsequently substituting the value of u in the equation 3.3 the y -coordinate of the rise curve corresponding to the base curve can be calculated. If it happens to lie in the straight line portion, then the value of y -coordinate can be directly computed from the equation 3.4 of the line. This procedure is repeated for all the points on the curve.

The shape design of the 3-D curve using the above approach can be presented in the form of a algorithm as below.

Algorithm:

Step1: Read the values of the end positions and tangent vectors r_A, r_B, t_A and t_B

Check whether $t_A \cdot t_B = 1$

If yes then the points r_A, r_B , and the tangents t_A and t_B form a plane.

else find the skew distance d

if skew distance $d = 0$

design a planar curve using the methodology presented in Chapter 2.

else Proceed to step 2.

Step 2: Set up O_1 -UVW coordinate system.

Step 3: Generate the base curve using the method presented in Chapter 2.

Step 4: Generate the rise curve.

Step 5: Design the space curve by combining step 3 and 4.

3.2 Spatial Trajectory Design

As mentioned in Chapter 2 trajectory planning involves the definition of time histories of time derivatives of position (position and orientation) vector in addition to the position vector itself. Knowing the expressions for r it is also essential to know the expressions for $\frac{dr}{dt}$ and $\frac{d^2r}{dt^2}$. It is essential to make a proper choice of the angles before one can define the orientation vector properly. The two angles chosen here are the tangent angle of the base curve represented by θ and the tangent angle of the rise curve represented by ψ . If s is the actual arc length of the 3-D curve and s' the arc length of the base curve of the corresponding point then, we can relate the first and second time derivatives of s and s' as follows.

$$s = s' \frac{s}{s'}$$

$\frac{s}{s'}$ can be assumed to be constant R and to be equal to the ratio of the arc length of 3-D curve to the arc length of base curve, therefore

$$\frac{ds}{dt} = \frac{ds'}{dt} R \quad (3.5)$$

and

$$\frac{d^2s}{dt^2} = \frac{d^2s'}{dt^2} R \quad (3.6)$$

Now $\frac{dr}{dt}$ and $\frac{d^2r}{dt^2}$ will be given by

$$\frac{dr}{dt} = \frac{dr}{ds'} \frac{ds}{dt} / R \quad (3.7)$$

$$\frac{d^2 \mathbf{r}}{dt^2} = \frac{\frac{d^2 \mathbf{r}}{ds'^2} \left(\frac{ds}{dt} \right)^2}{R} + \frac{\frac{d\mathbf{r}}{ds'} \frac{d^2 s}{dt^2}}{R} \quad (3.8)$$

Knowing all the above terms on the right hand side of the equations (3.7) and (3.8), the variation of $\frac{d\mathbf{r}}{dt}$ and $\frac{d^2 \mathbf{r}}{dt^2}$ as functions of s are calculated. We also know the variation of s as a function of time t . From these relations $\frac{d\mathbf{r}}{dt}$ and $\frac{d^2 \mathbf{r}}{dt^2}$ can be expressed as functions of time t . The x , y and z components of $\frac{d\mathbf{r}}{ds}$ and $\frac{d^2 \mathbf{r}}{ds^2}$ will be:

$$\frac{dx}{ds'} = \cos[\theta(s')] \quad (3.9)$$

$$\frac{dy}{ds'} = \sin[\theta(s')] \quad (3.10)$$

$$\frac{d^2 x}{ds'^2} = -\sin \theta(s') \frac{d\theta(s')}{ds'} \quad (3.11)$$

$$\frac{d^2 y}{ds'^2} = \cos \theta(s') \frac{d\theta(s')}{ds'} \quad (3.12)$$

where

$$\theta(\sigma) = \int_0^\sigma k(s') ds' + \theta_0 \quad (3.13)$$

For the z -component we have two models of the rise curve, the clothoid and the cubic models. The expressions for each model are given separately below.

- For clothoid model

$$\frac{dz}{ds'} = \tan[\psi(s')] \quad (3.14)$$

where s' is the arc length of the rise curve in the w - s plane and $\psi(s')$ is the tangent angle of the rise curve.

similarly

$$\frac{d^2 z}{ds'^2} = \sec^2[\psi(s')] \quad (3.15)$$

where

$$\psi(\sigma) = \int_0^\sigma k(s') ds' + \psi_0 \quad (3.16)$$

- For a cubic curve these equations are straight forward. The computational complexity with cubic blended rise curve is also very less. These equations for the cubic curve will take the form as follows:

$$\frac{dw}{ds'} = \frac{dw/du}{ds'/du} \quad (3.17)$$

Again differentiating the above equation with respect to s' and simplifying we get

$$\frac{d^2 w}{ds'^2} = \frac{\frac{dw/du}{ds'/du} \frac{ds'^2}{du^2} - \frac{d^2 w}{du^2}}{(ds'/du)^2} \quad (3.18)$$

The expressions for all the right hand side terms are known in a closed form and they are quite simple to compute for different values of u which makes the calculation of $\frac{d^2 w}{ds'^2}$ quite simple as compared to the above method. An illustrative example of 3-D trajectory planning for C-S-C shape model base curve and cubically blended rise curve, is considered in the next Chapter.

3.3 Applications of Spatial Trajectory

One of the most potential application is for planning the path of mobile robots (or AGVs) where the motion is in three-dimensional space (ie) the platform of the robot has to move not only in a plane parallel to the ground but also has to take a lift in the direction perpendicular to the ground plane. These curves designed can also be used to plan the trajectory of the manipulator arm in cartesian space.

The space curves are used to design the connecting road between the two highways or flyover where there is a difference of height at both the ends. The use of intrinsic geometry approach not only allows the designer to evaluate the parameters such as curvature explicitly, but also provides the flexibility to change the shape of the path fulfilling the end point conditions. This

can be achieved by choosing a different shape model and also by altering the value of the sharpness λ_d .

CHAPTER 4

CASE STUDIES AND EXAMPLES

4.1 Implementation Details

The programs developed for the design of curves and the trajectory are all modular and interactive. The graphics hardware facility used for this purpose are a SUN - 3/60 workstation and the software infrastructural utilities such as the Sunview are used. The program is menu driven for input purposes and also offers the choice of the final output to be viewed on the screen. All the modules have been written in 'C' language. These programs can be ported on to any system with the required changes in the graphics module, which is implementation dependent. For the purpose of hard copy output AutoCAD has been used which uses the program output as input data.

4.2 Examples of 2-D Curve Design

As discussed in Chapter 2, two shape models for planar curves (S-C-S and C-S-C) are considered for this study. Specific numerical examples for both the models are discussed in this section with a typical example of each being considered for the time differentials of position vector, which makes the discussion on 2-D trajectory design complete in its sense.

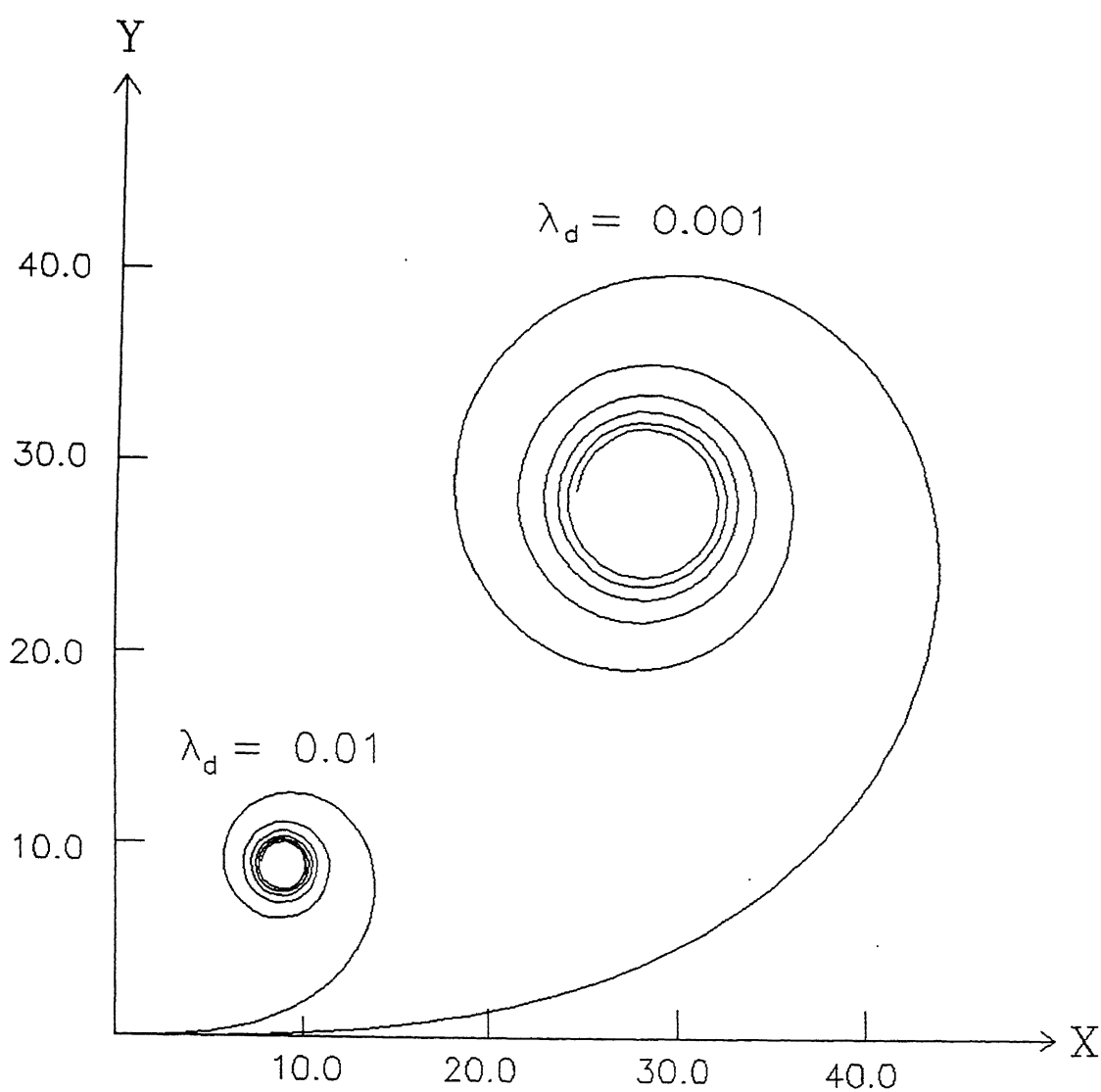


Figure 4.1 Cartesian Plot of Cornu Spirals with different sharpness

To have an idea as to how the clothoid curve changes in size with change in the given sharpness λ_d , the cartesian output of two clothoid curves (cornu spirals) with $\lambda_d = 0.01$ and $\lambda_d = 0.001$ is shown in Figure 4.1. From these figures it is evident that as the value of λ_d increases the curve becomes more sharp or the increase of curvature is fast

Illustrative Examples of S-C-S Shape Model Curves

Two types for S-C-S shape model curves have been discussed in Chapter 2. They are

1. S-C-S shape model curve with positive change in tangent angle.
2. S-C-S shape model curve with negative change in tangent angle.

Figure 4.2 shows the cartesian space of the curves with different sharpness of $\lambda_d = 0.01$, $\lambda_d = 0.001$ and $\lambda_d = 0.0002$ for S-C-S shape model with positive tangent angle change. It can be observed that as the value of the sharpness λ_d decreases the exit distance of the curve increases. The maximum exit distance is $\min(l(P_0, Q), l(P_1, Q))$ as shown in Figure 2.9. If the value of λ_d specified is less than that value required for maximum distance then it is not possible to design a curve for that given value of λ_d . In the program developed the user is prompted to make a choice, either to reduce the value of λ_d or to let the computer find out the minimum value of λ_d and design the curve. Figure 4.3 shows the curves again with the same sharpness and the end position vector conditions of $P_0 = (25,25)$ and $P_1 = (125,125)$ but

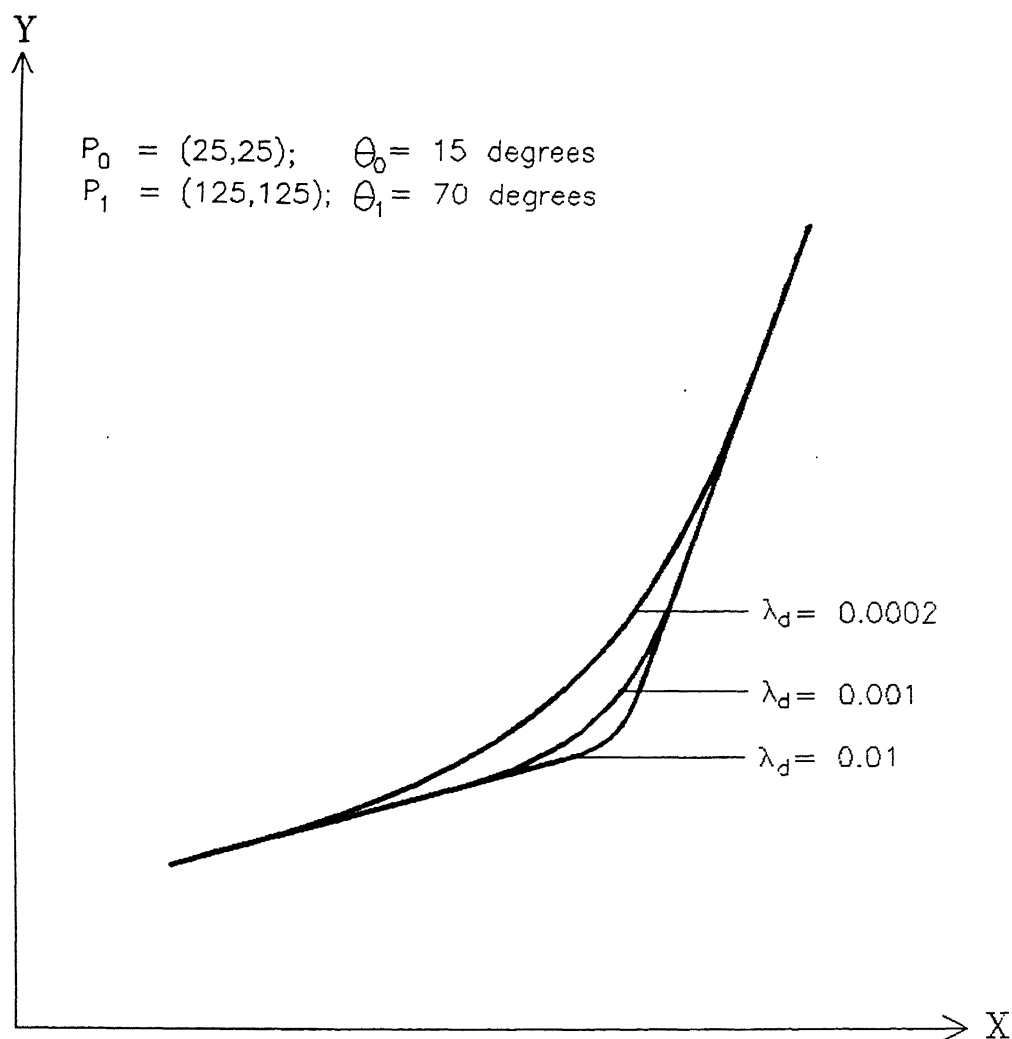


Figure 4.2 Example of S-C-S Shape Model Curve with Positive Change in Tangent angle

the change in tangent angle in this case is negative. The end tangent conditions are different, with $\theta_0 = 70$ degrees and $\theta_1 = 15$ degrees, the variation of values for λ_d is also same.

Illustrative Examples of C-S-C Shape Model Curves

The different types of C-S-C shape model curves possible are

1. Tangent angle change for the first clothoid pair is positive and that for the second clothoid pair is negative.
2. Tangent angle change for the first clothoid pair is negative and that for the second clothoid pair is positive.
3. Tangent angle change for both the clothoid pairs is positive.
4. Tangent angle change for both the clothoid pairs is negative.

Figure 4.4 shows C-S-C shape model curve in which the tangent angle change for the first clothoid pair is positive and for the second clothoid pair is negative. The end position vector conditions are kept the same for (ie) $P_0 = (25,25)$ and $P_1 = (125,125)$, and the tangent angles are $\theta_0 = 0$ and $\theta_1 = 0$ degrees. Three curves with different values of $\lambda_d = 0.01$, $\lambda_d = 0.002$ and $\lambda_d = 0.0015$ are shown. It can be observed that as the value of λ_d decreases the curve length increases and the curve deviates more and more from the straight line joining the two points. The slope of the intermediate straight line portion also increases with the decrease in the value of λ_d .

Figure 4.5 shows an illustrative example of the C-S-C shape

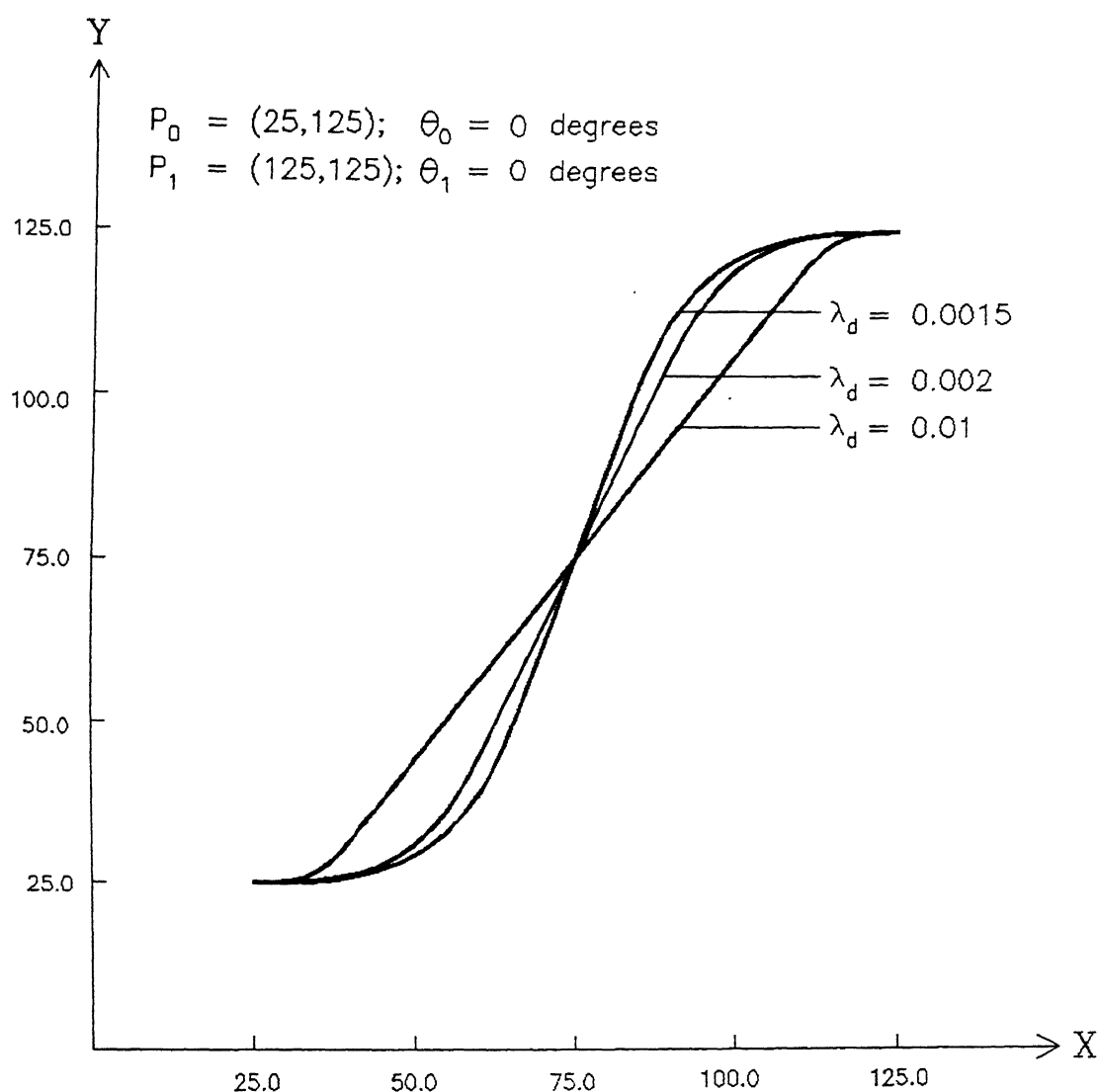


Figure 4.4 Example of C-S-C Shape Model Curve with Positive Change in Tangent angle for the First CLothoid Pair and Negative for Second

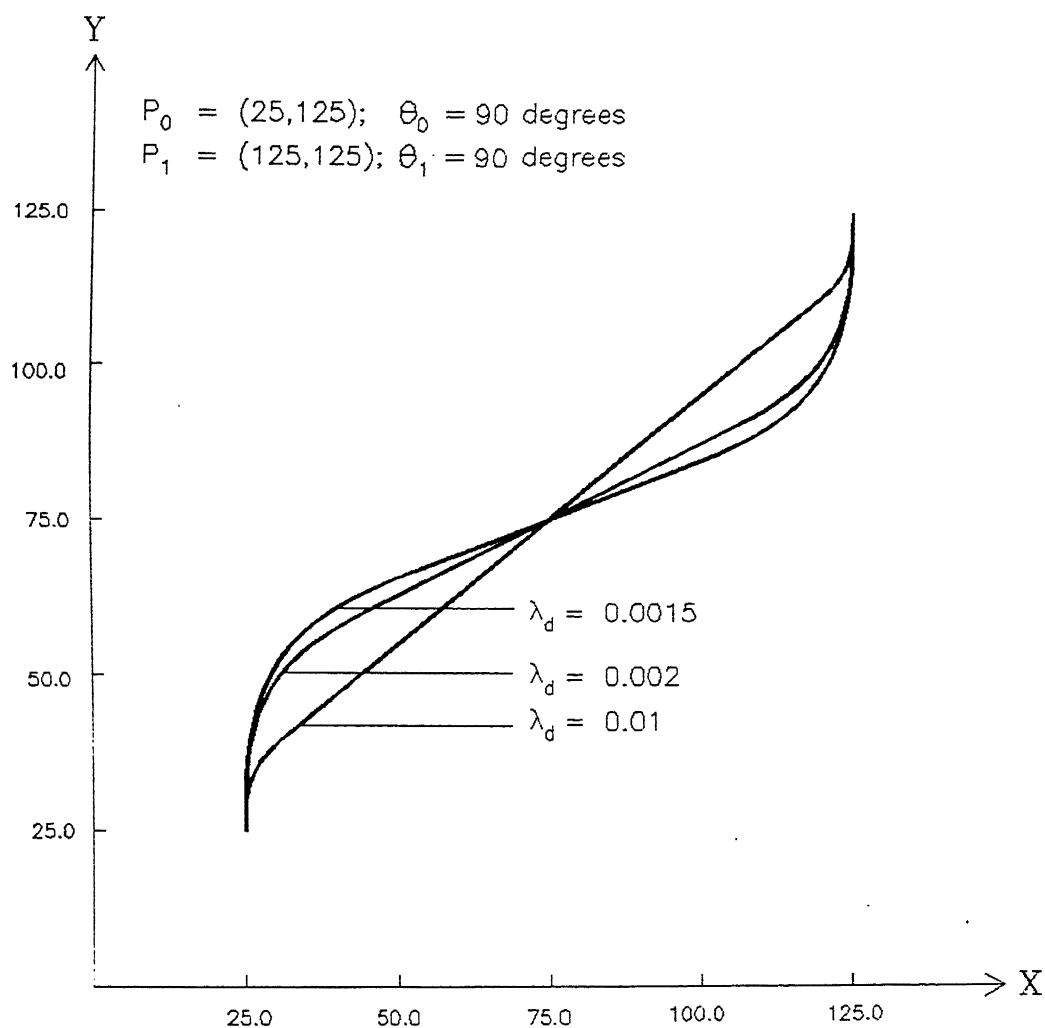


Figure 4.5 Example of C-S-C Shape Model Curve With Negative Change in Tangent Angle For the First Clothoid Pair and Positive for the Second

model curve with the same end position vector conditions but different tangent angles $\theta_0 = 90$ and $\theta_1 = 90$ degrees. With these end tangent conditions the curve designed has a negative change in the tangent angle for the first clothoid pair and positive for the second clothoid pair. For this case also the same variation of the sharpness λ_d is shown. For both the above cases the net change in the tangent angle over whole of the curve is zero and hence the change in tangent angle for the first clothoid pair is equal and opposite to that of the second clothoid pair.

Figure 4.6 shows the example of C-S-C shape model curve in which the tangent angle change for both the clothoid pairs is positive. The end point conditions are again kept the same with the end tangent conditions being changed to $\theta_0 = 0$ and $\theta_1 = 90$ degrees. Unlike in the above two cases it is observed that the variation in the value of λ_d does not affect the slope of the intermediate straight line portion, whereas the length of the curve increases as normal with the decrease in the sharpness λ_d .

Figure 4.7 shows the example of C-S-C shape model curve in which the tangent angle change for both the clothoid pairs is negative. The end point conditions are again kept the same with the end tangent conditions being changed to $\theta_0 = 90$ and $\theta_1 = 0$ degrees. As in the above case it is observed that the variation in the value of λ_d does not affect the slope of the intermediate straight line portion, whereas the length of the curve increases

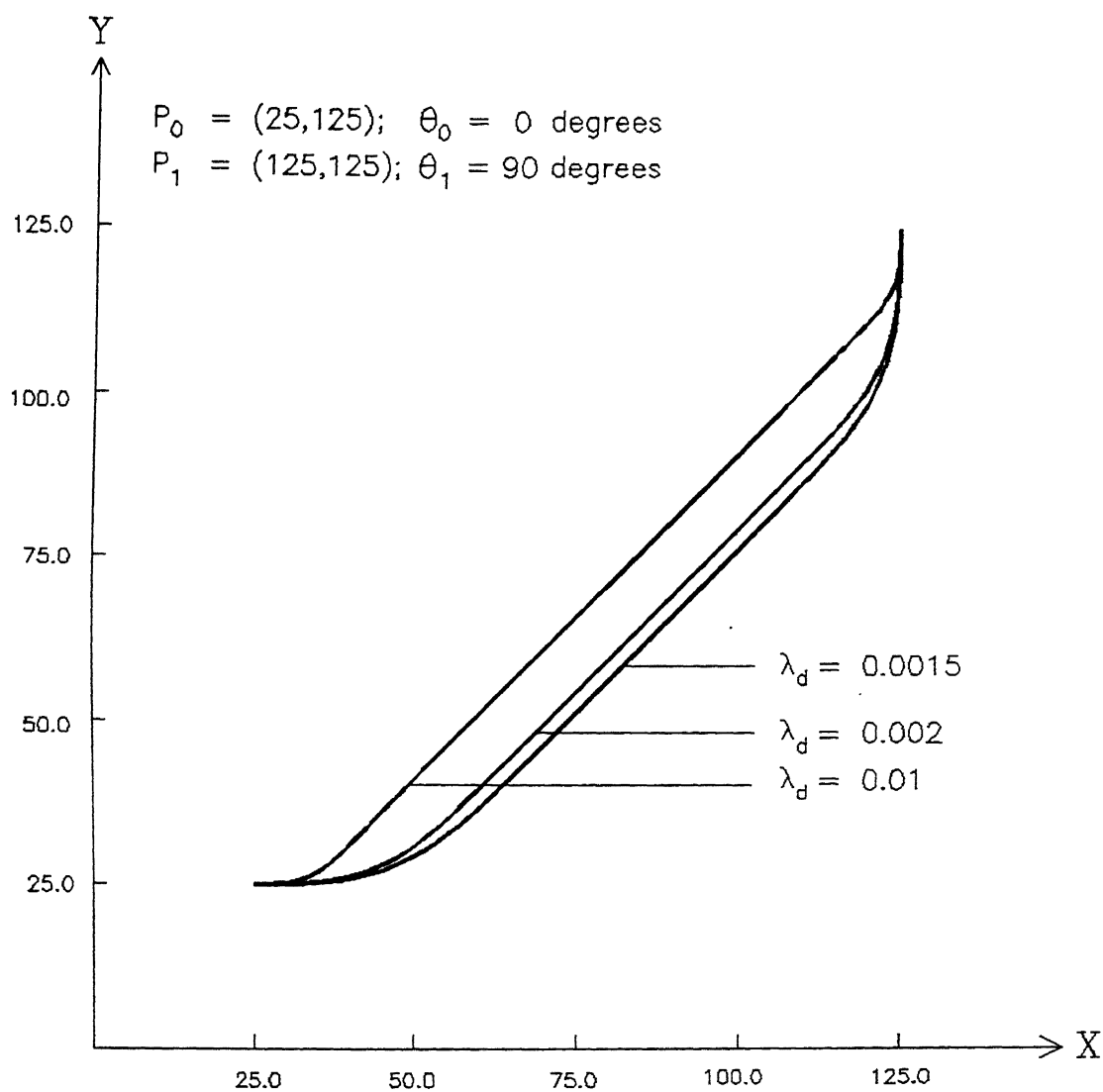


Figure 4.6 Example of C-S-C Shape Model Curve with Positive Change in Tangent angle for the for both the Clothoid Pairs

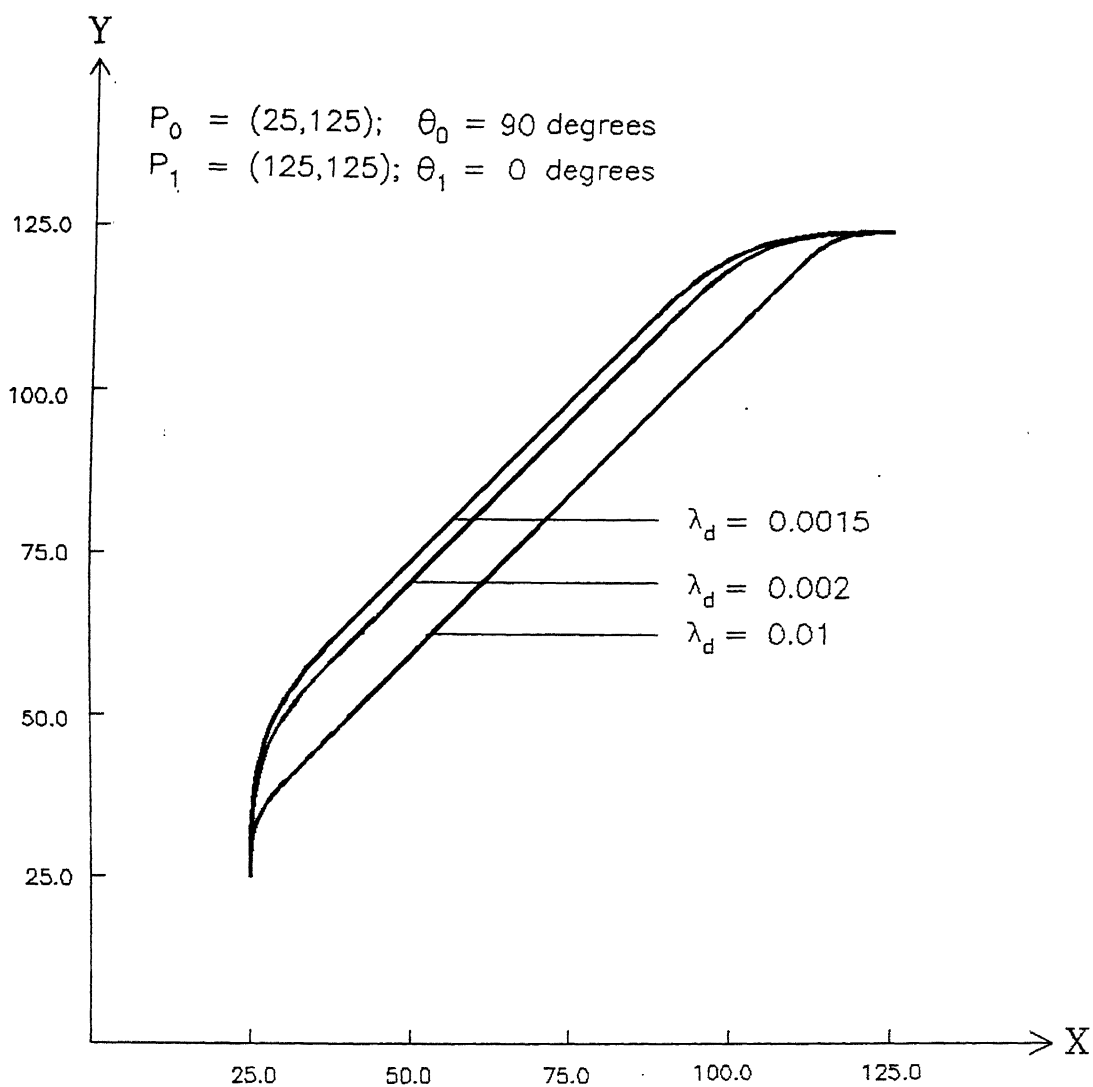


Figure 4.7 Example of C-S-C Shape Model Curve with Negative Change in Tangent Angle For the both the Clothoid Pairs

as normal with the decrease in the sharpness λ_d . In these two types of models where the tangent angle change is either negative or positive for both the clothoid pairs the, algorithm for the calculation of the slope of intermediate straight line portion converges very fast. It can also be noted that the slope of the straight line portion in between is not a function of the sharpness λ_d but depends only on the end tangent conditions. For this model also there is a minimum value of λ_d , for which the curve can be designed, if the value of λ_d is further reduced then the designed curve would make a loop which is not considered here. The algorithm described for calculating the slope of the intermediate straight line portion takes care for eliminating these cases where the curve makes a loop. The algorithm described for the design of C-S-C shape model curves takes care of all the four types depending on the input data. All the examples considered here are in the first quadrant only, if the input data given is in the other quadrants then the required transformation of the axis system has to be done at the appropriate place. The algorithm described takes care for these cases in other quadrants also.

To make the discussion on trajectory planning complete it is necessary to know the variation of velocities and acceleration as a function of time. An example of the C-S-C model curve with positive change in tangent angle for the first clothoid pair and negative for the second, with the sharpness $\lambda_d = .002$ is

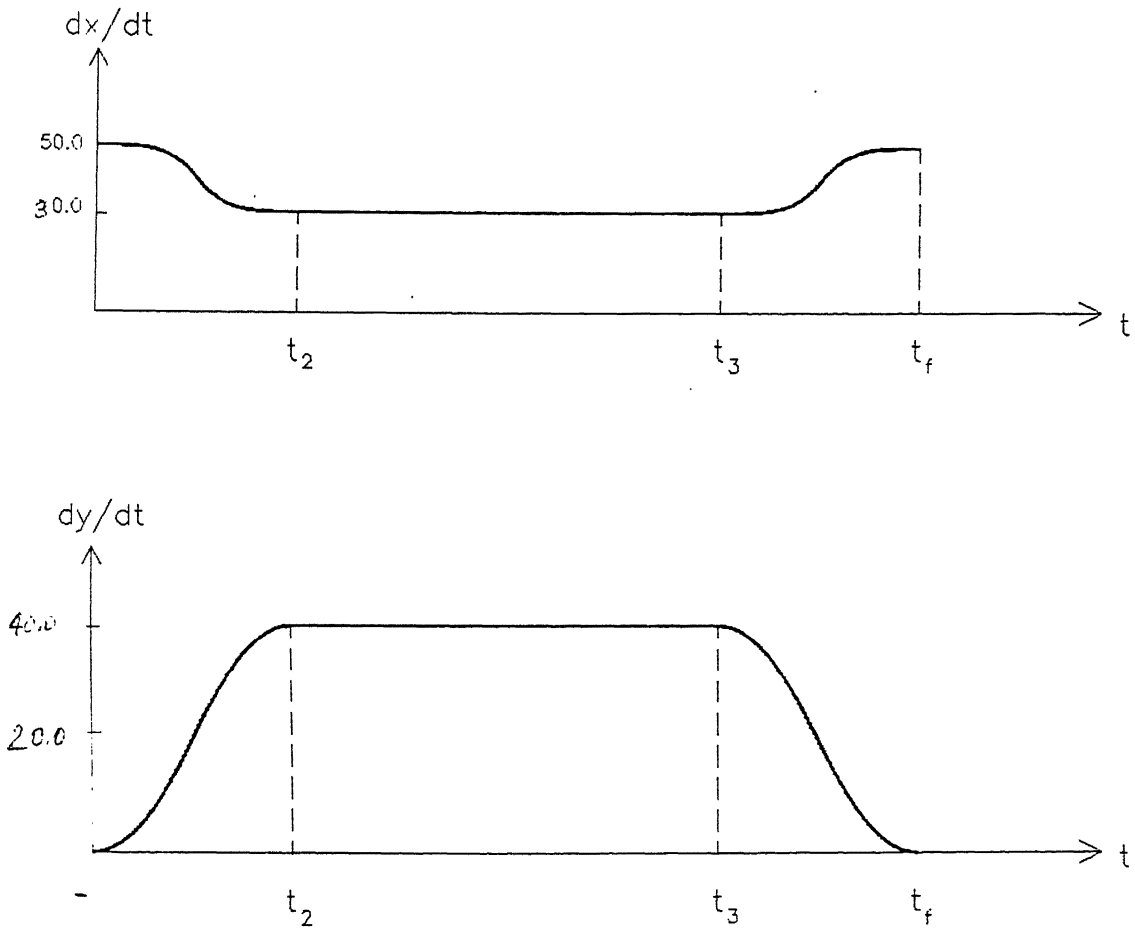


Figure 4.8 Velocity Plots for C-S-C Shape Model Curve with $ds/dt = \text{constant}$.

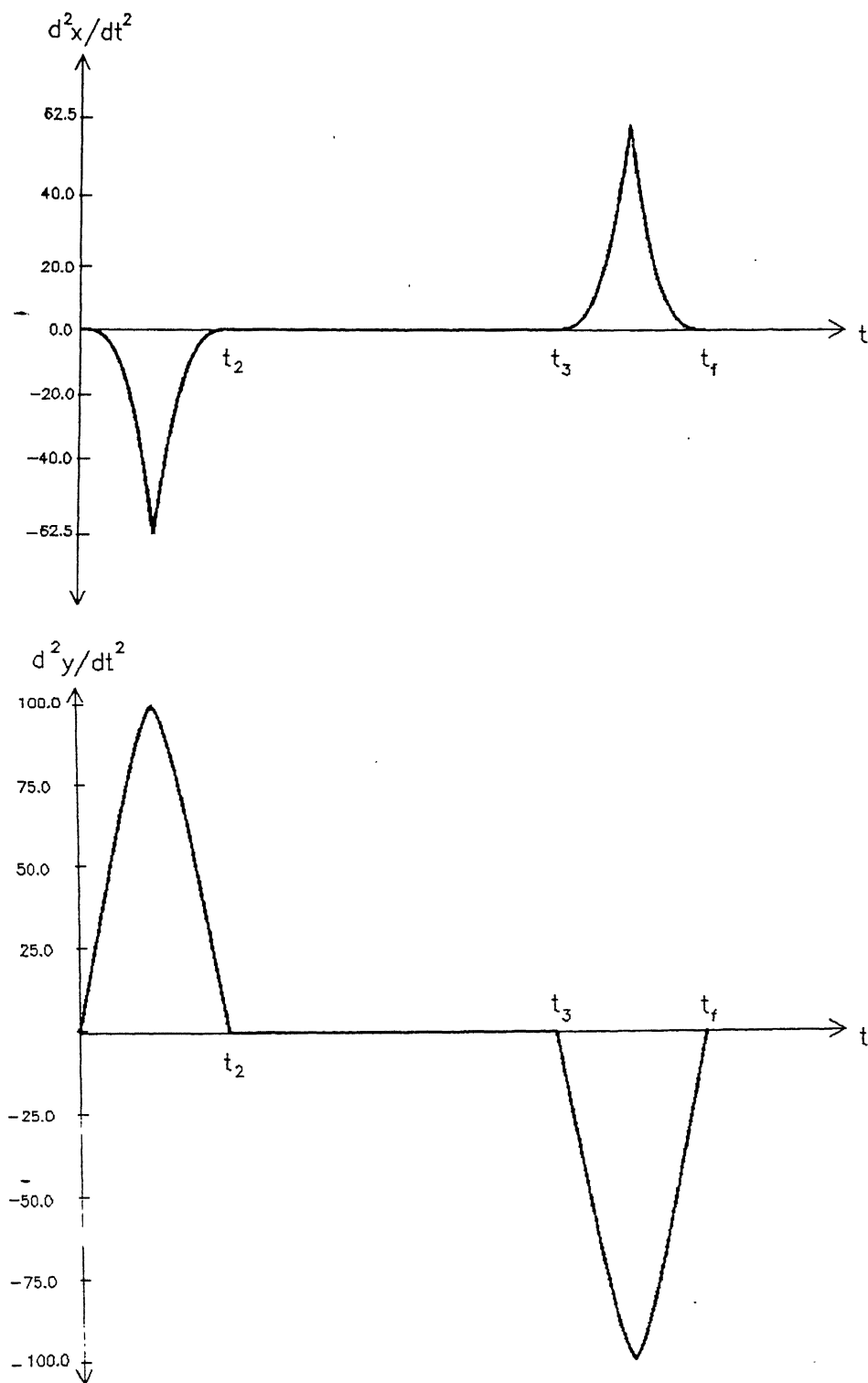


Figure 4.9 Acceleration Plots for C-S-C Shape Model
Curve with $ds/dt = \text{constant}$

considered. One more input is required to plan the trajectory that is the speed vs. time relationship. Once this relationship is specified as a closed form function or some numerical discrete data points being provided we can now calculate dx/dt and dy/dt from equation (2.36). To simplify the problem ds/dt is taken to be constant over the whole period of time. In the example the value of ds/dt is taken as 50 representative units. The velocity and acceleration plots for the x and y directions are shown in Figures (4.8) and (4.9). This example where $ds'/dt = \text{constant}$ can serve as a good model to plan the path of a mobile robot in 3-D when the robot platform travels at a constant speed all over the path length.

4.3 Examples of 3-D Curve Design

For a 3-D curve design the input variables are again the same (ie) P_0 , P_1 , t_0 and t_1 , but these are vectors of dimension 3 by 1, the x, y and z components of these vectors have to be specified. For the example considered here $P_0 = (25, 25, 25)$, $P_1 = (125, 125, 125)$, $t_0 = (1, 0, 0)$ and $t_1 = (1, 1.713, 0)$, the value of λ_d being equal to 0.002. These tangent vectors t_0 and t_1 are not the normalized vectors. The curve is designed on the lines discussed in Chapter 3. The base curve is modeled as a C-S-C shape model curve, to model the rise of the curve the cubically blended curve discussed in Chapter 3 has been used. Figure 4.10 shows the isometric view of the 3-D curve. Figure 4.11 shows the base curve in the U-V plane and the rise curve in the W-S' plane. As an

$$P_0 = (25, 25, 125); t_0 = (1, 0, 0)$$

$$P_1 = (125, 125, 125); t_1 = (1.7, 1.0)$$

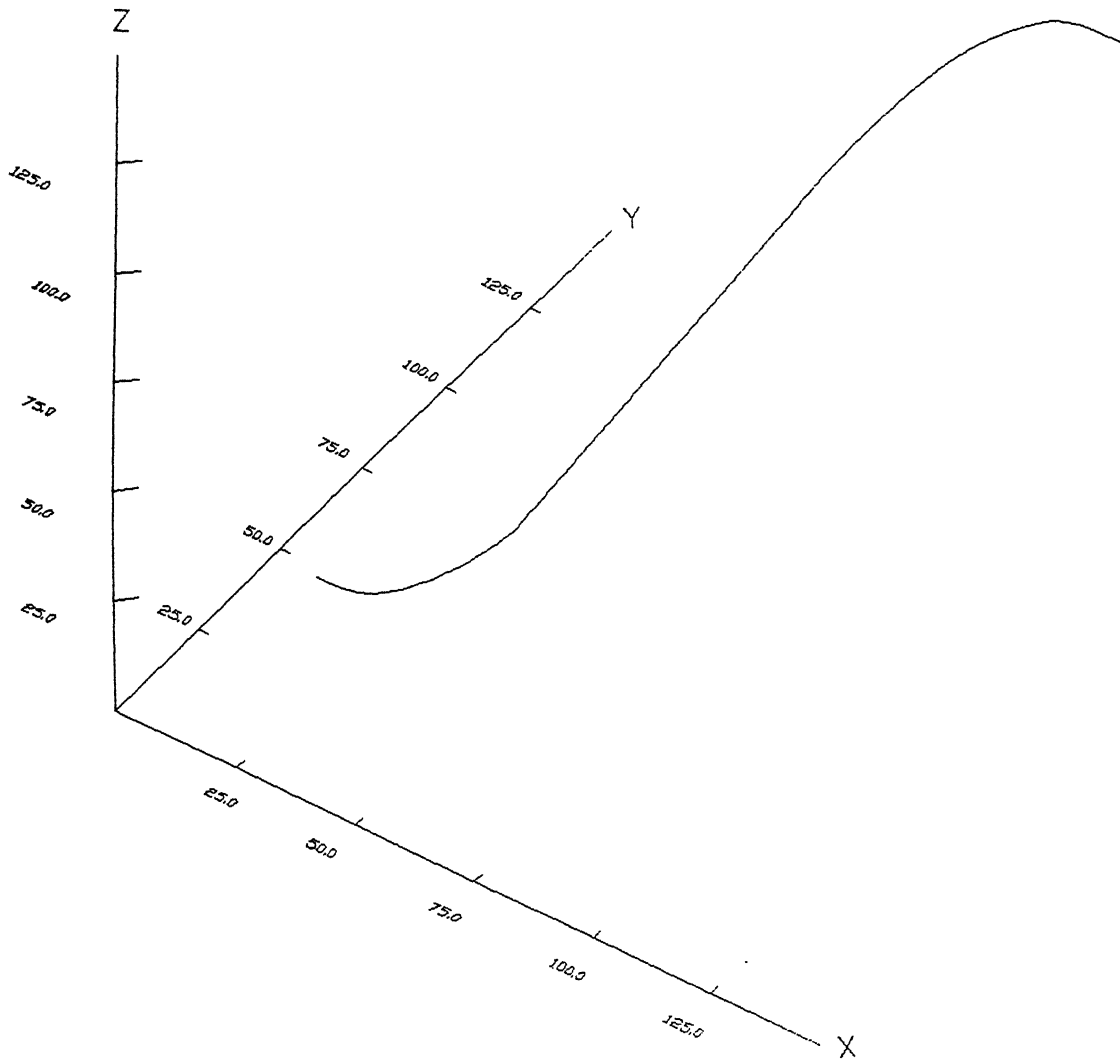
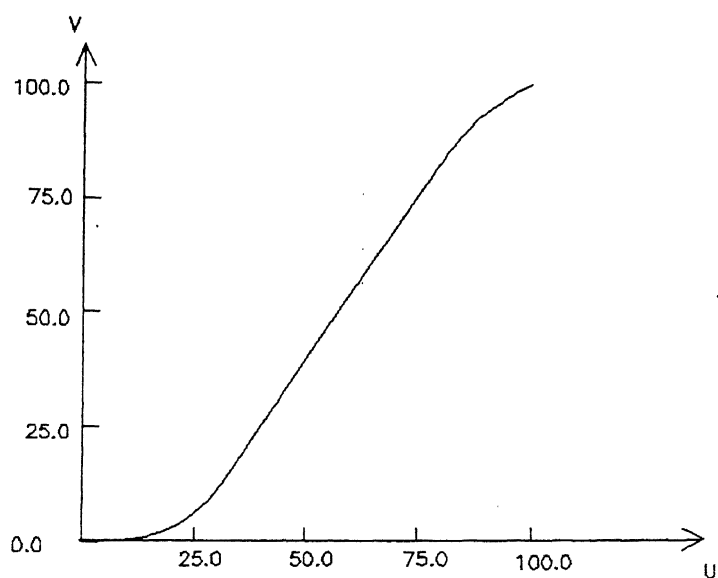
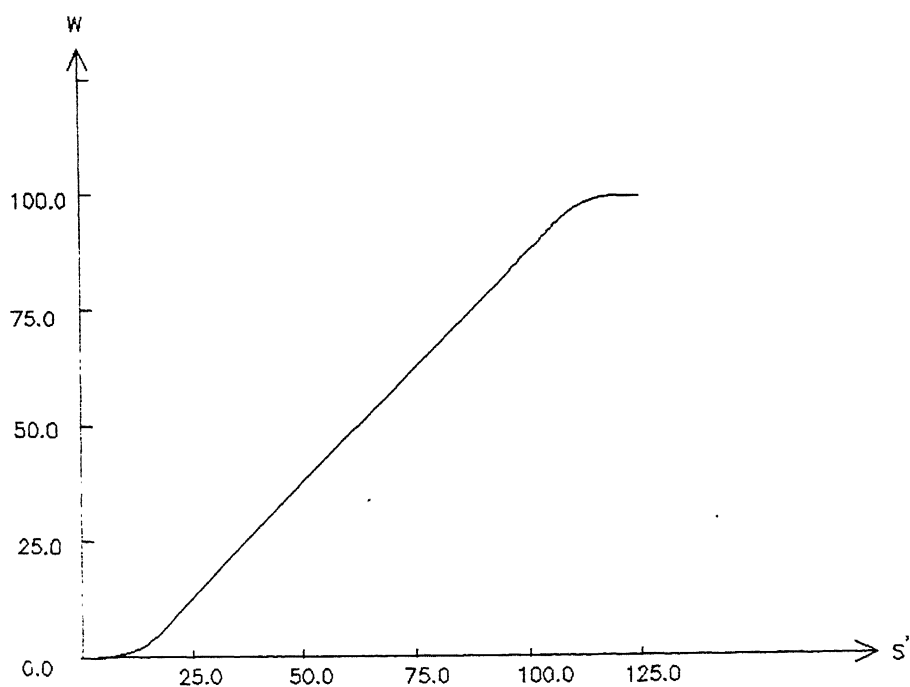


Figure 4.10 Isometric View of C-S-C Shape Model
Curve with Cubically Blended Rise Curve



$U - V$ Plane



$W - S$ Plane

Figure 4.11 Base Curve with C-S-C Shape Model and Rise Curve with Cubic Blending for the 3-D Curve

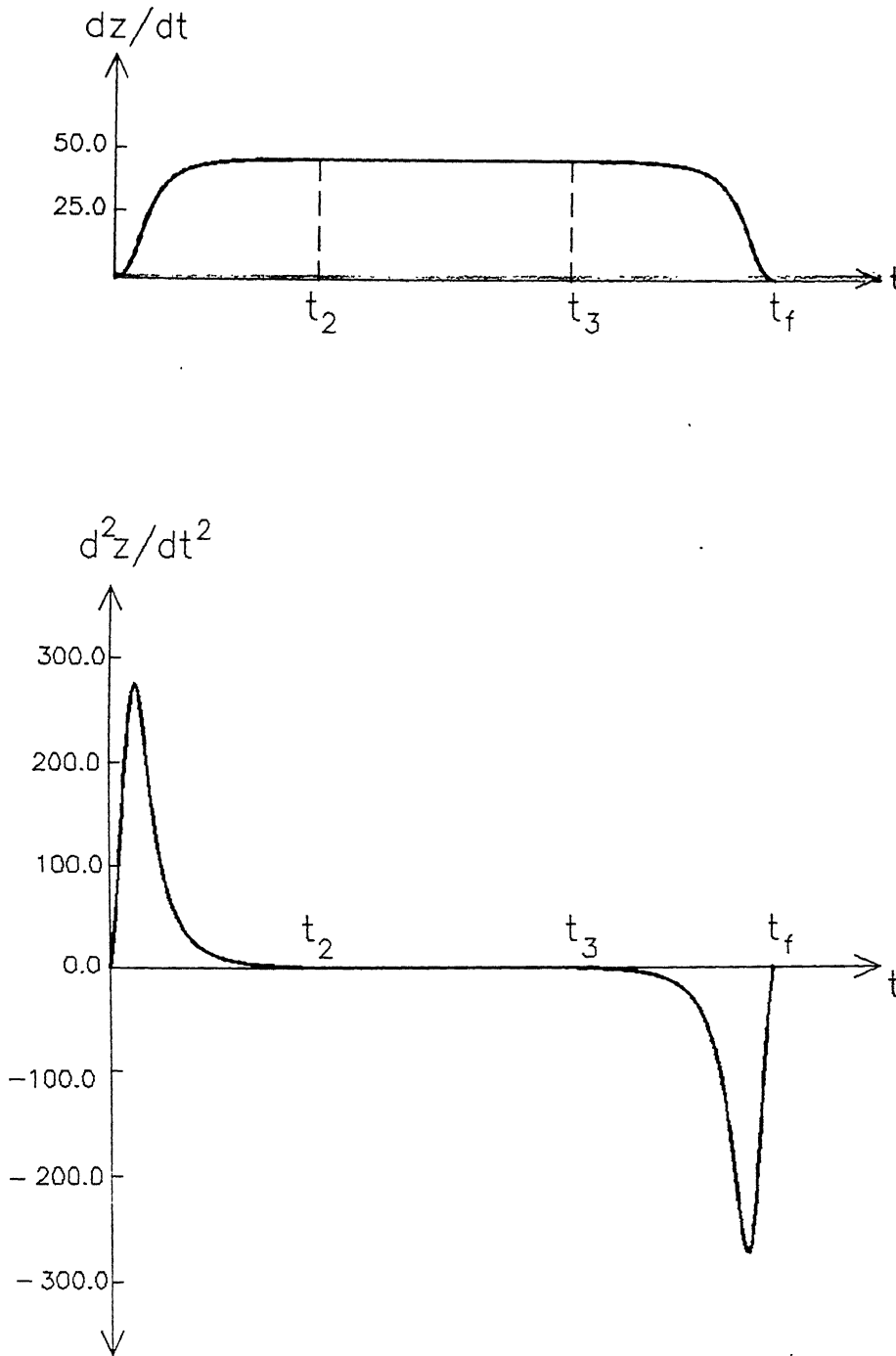


Figure 4.12 Velocity and Acceleration Plot of the Z-Coordinate, when the Rise Curve is Cubically Blended

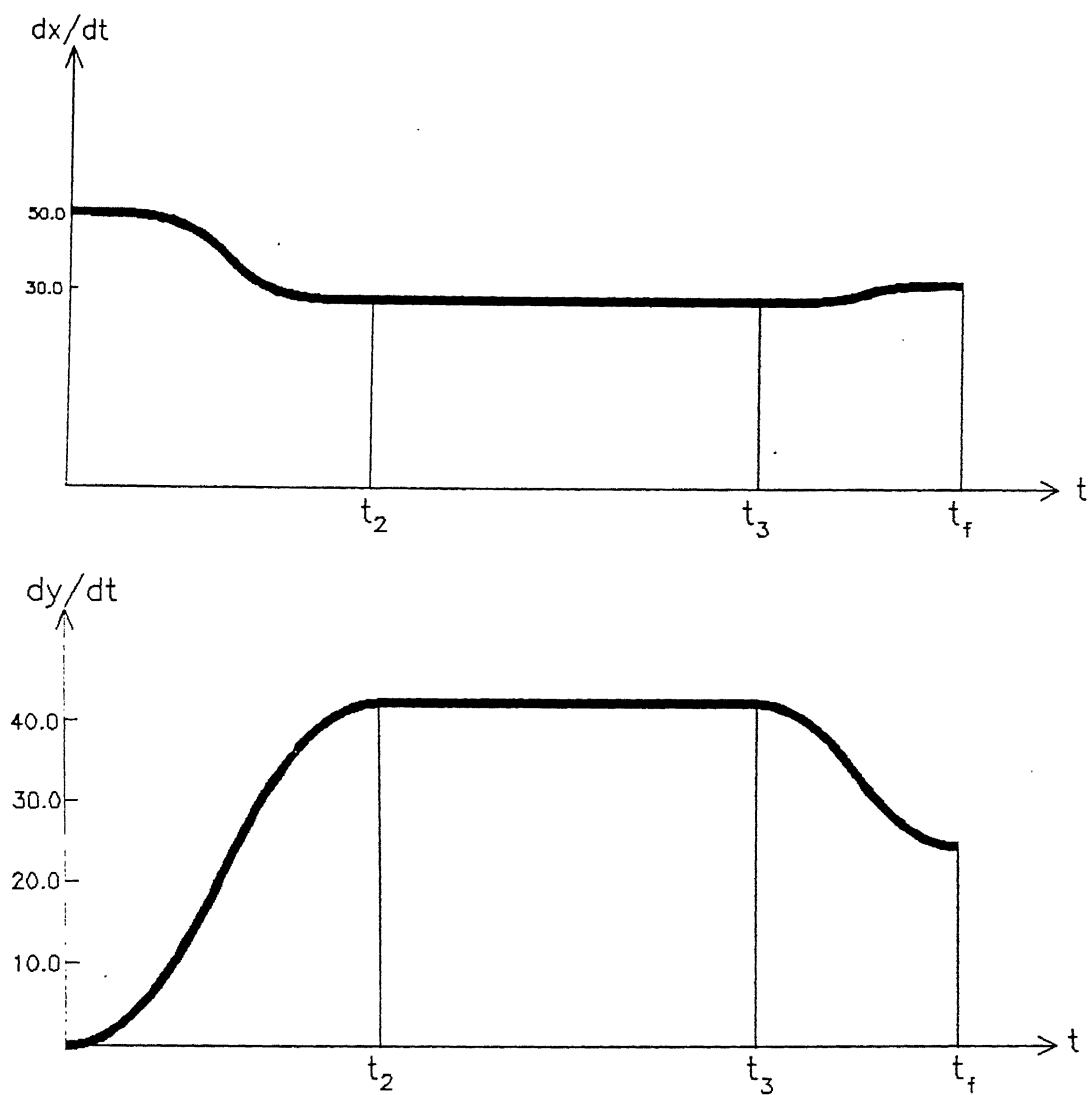


Figure 4.13 Velocity Plots for 3-D Curve Example with C-S-C Shape model for Base Curve

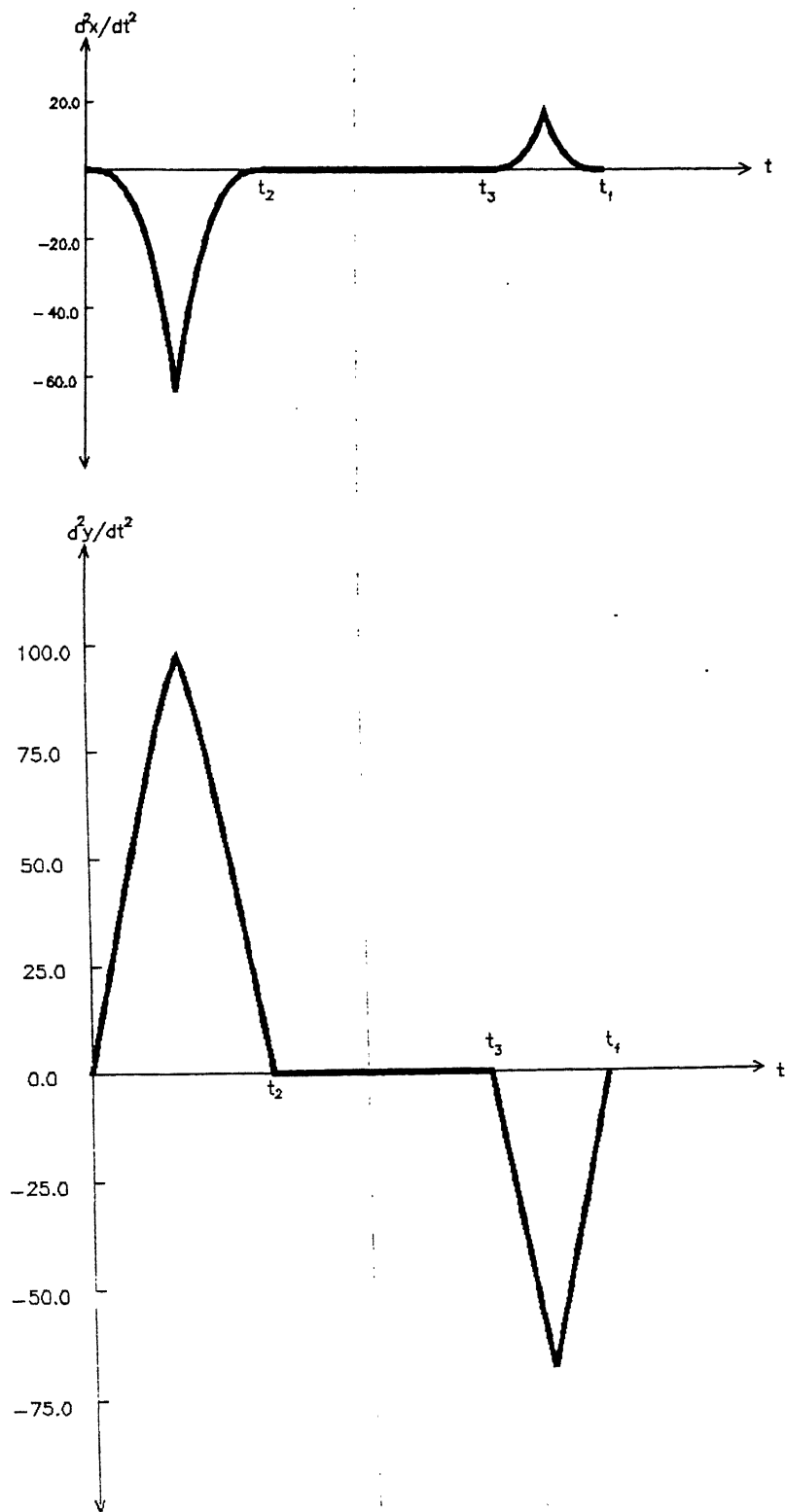


Figure 4.14 Acceleration Plots for 3-D Curve Example
with C-S-C Shape model for Base Curve

example for the trajectory planning problem, the same above problem is considered here. The value of ds'/dt taken is again assumed 50 representative units. For the base curve of this 3-D curve the change in tangent angle at both the clothoid pairs is different, and hence the maximum curvature for the clothoid pairs is different, consequently for $ds/dt = \text{constant}$, the maximum accelerations will also be different at both the clothoid pairs unlike in the previous example for 2-D where the maximum accelerations are same. It can also be observed that a component of the x and y velocities remain at the end position. The velocity and acceleration plots are shown in Figure 4.13 and 4.14. The plots of z component of velocity and acceleration are shown in figure (4.12) it can be noted that the acceleration is not evenly distributed over the whole of the time period $[0, t_2]$ and this makes it shoot to higher values than in the clothoid blended curves. Moreover it is very difficult to control the maximum acceleration directly, where as in case of clothoid blending it can be directly controlled by controlling the curvature. This is the essence of the present work.

4.4 Observations On Numerical Work

The integrals (Fresnel Integrals) to be evaluated for calculating the cartesian coordinates are evaluated numerically at small intervals of the arc length s. Evaluation of this integral is the most compute intensive job in whole of the program. In the design of C-S-C shape model curve this has to be evaluated quite a

large number of times in the iterative procedure for calculating the tangent angle of the intermediate straight line portion.

For the purpose of display of a 3-D curve points for isometric view, U-V plane and W-S' plane are computed using the appropriate transformation matrices. Care has to be taken while selecting the view point for the isometric view. The direction cosines of the line joining the initial and final points should not be the same or proportional to the view point coordinates, in which case the curve will always appear to be a closed curve.

A number of plots has to be displayed for the complete information of the curve and the trajectory design (velocity and acceleration plots in the x y and z direction.

CHAPTER 5

CONCLUSIONS

5.1 Technical Summary

A variety of parametric and non-parametric curve designing techniques are available for the design of two-dimensional and three-dimensional curves. These techniques lack in controlling the intrinsic properties of curves directly. A methodology of shape design using intrinsic geometry has been proposed for two-dimensional and three-dimensional curves. Two shape models (S-C-S and C-S-C) have been proposed for the design of two-dimensional curves, using piece-wise linear and continuous curvature elements which ensure C^2 continuity of the curve in cartesian space. For the design of three-dimensional curves a pseudo-intrinsic method is discussed in which the three-dimensional curve is designed by combining the definitions of two planar curves, the base curve and the rise curve. The base curve is defined using the concepts of intrinsic geometry as above, and the rise curve is a cubically blended curve with a straight line portion in between the two end cubic portions or also it can be designed modeling it as a C-S-C shape model curve. The base curve is defined in a plane parallel to both the tangent directions and the rise curve in the direction parallel to the skew direction. Finally the three-dimensional curve is defined as

a parametric function of the base curve arc length s' .

Conventional trajectory planning schemes generally "interpolate" or "approximate" the desired path by a class of polynomial functions or by some parametric curves, and generates a sequence of time based "control set points" for the control of manipulator arm, from the initial location to its destination. In these methods a direct control over the variables like velocity and acceleration is not possible. The curves designed using intrinsic geometry makes it is possible to exercise a control over the variables like velocity and acceleration. The intrinsic properties can be controlled directly when the curves are designed using the concepts of intrinsic geometry, these can be associated to the velocities and acceleration if these curves are made use of in planning the trajectories. This concept can be applied in planning the trajectories to overcome the short-coming of the conventional methods.

5.2 Recommendations for Further Work

Intrinsic geometry approach for design of curves requires mapping of data from the intrinsic space to the cartesian space. This mapping is accomplished through Serret-Frenet equations which are a set of coupled differential equations. In the present work linear curvature elements have been considered. To solve this we land up with the Fresnel integrals, which are evaluated numerically. The series summation approach has been used to

evaluate these integrals, which makes it a compute intensive job. It is however desirable to develop a computationally efficient algorithm or procedure for the solution of Serret-Frenet equations. Since the range of values required is not very large and normally required in the range $[0,\pi]$ one can also make use of look-up tables at the cost of memory of the computer.

In the present work the curvature function $k(s)$ is assumed to be a set of contiguous linear curvature elements. Instead this work can be tried out using a set of cubic spirals. A cubic spiral is a curve whose tangent direction is described by a cubic function of path distance s . This approach is expected to give more smoother paths as compared to the one with linear curvature functions, at the cost of increased computational complexity (Kanayama and Hartman, 1988).

REFERENCES

- [1]. Brady, M., Hollerbach, J.M., T.L.,Lozano-perez, T., and Mason, M.T.,
Robot Motion: Planning and Control,
The MIT Press, Cambridge, Massachusetts.

- [2]. Farin, G.,
Curves and Surfaces for Computer Aided Geometric Design - A Practical Guide,
Academic Press, Inc., San Diego, 1988.

- [3]. Faux, I.D., and Pratt, M.J.,
Computational Geometry for Design and Manufacture,
Halsted , New York, 1979.

- [4]. Fu,K.S., Gonzales,R.C., and Lee,C.S.G.,
Robotics: Control, Sensing, vision, and Intelligence,
McGraw-Hill Book Company, New York, 1987.

- [5]. Hsai,L.M. and Yang,A.T.
"On the Intrinsic Properties of Point Trajectories in
Three-Dimensional Kinematics"
" ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION
IN DESIGN, Sept., Vol. 107, No.4, 1985, pp. 401-405.

- [6]. Kanayama,Y., and Miyake,N.,
"Trajectory Generation for Mobile Robots,"
Robotics Research, Vol. 3. MIT Press, pp. 333-340, 1986.
- [7]. Kanayama,Y., and Hartman,B.I.,
"Smooth Local Path Planning for Autonomous Vehicles,"
Autonomous Robot Vehicles edited by I.J.Cox and G.T.Wilfong
Springer Verlag, pp. 63-67.
- [8]. Kreyszig, E.,
Differential Geometry,
Mathematical Expositions No. 11, University of Toronto Press,
Toronto, 1959.
- [9]. Mortenson,M.E.,
Geometric Modeling,
John Wiley and sons, New York, 1985.
- [10]. Paul,R.P.,
"Manipulator Cartesian Path Control",
IEEE Trans. System, Man, Cybern, Vol. SMC-9, No. 11,
pp. 702-711, 1979.
- [11]. Rogers,D.F., and Adams,J.A.,
Mathematical Elements for Computer Graphics,

McGraw-Hill Book Company, New York, 1990.

[12]. Ryuth,B.S., and Pennock,G.R.,

"Accurate Motion of a Robot End-Effector using the Curvature Theory of a Ruled Surface,

" ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION IN DESIGN, Dec., Vol. 110, No.4, 1988, pp. 383-388.

[13]. Ryuth,B.S., and Pennock,G.R.,

"Trajectory Planning Using the Ferguson Curve Model and Curvature Theory of a Ruled Surface,"

ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION IN DESIGN, Sept., Vol. 112, No.4, 1990, pp. 377-383.

[14]. Shahriar, T.,

"Shape Design Using Intrinsic Geometry"

Ph.D. Thesis submitted to Department of Mechanical Engineering, V.P.I & S.U., Blacksburg, Virginia, 1991.

[15]. Struik,D.J.,

Lectures on Classical Differential Geometry

Addison Wesley Press, Inc., Cambridge 42, Mass 1950.